

Publicación práctica para usuarios de

commodore

Noviembre 1985 Precio 350 Ptas Año 1 Número 3

¿COMMODORE O SINCLAIR?

BUSCANDO EL MEJOR CAMINO

ACELERA TUS RUTINAS DE JUEGO

CONTIENE UN POSTER


BIENVENIDOS A LA INGLATERRA FEUDAL
The EVIL Crown
Venciendo en los torneos reales podrás llegar a ser Rey.
¿Podrás apoderarte del resto del mundo medieval y adueñarte de la corona malvada?
APARICION SIMULTANEA EN INGLATERRA Y EN ESPAÑA
SPECTRUM 48
COMMODORE
LA CORONA MALVADA: UN JUEGO DE SIMULACION ANIMADA. LAS FUERZAS POLITICAS EN LA EDAD MEDIA. ¡FACIL DE JUGAR, MUY DIFICIL DE GANAR!
MIND GAMES
PREMIUM SOFTWARE

WER

RE, S.A.

s. 232 24 61 - 232 25 52
ONA (SPAIN)

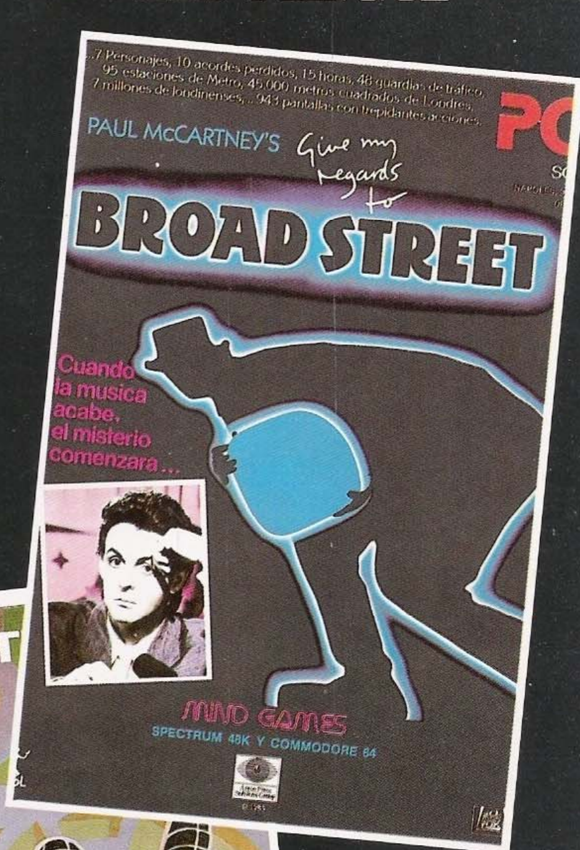


SOFTWARE

CENTURY
SOFTWARE

¡¡¡ATENCION!!!

POWER, SOFTWARE, S.A. Perseguirá por todos los medios legales a su alcance, cualquier tipo de piratería, comercialización, importaciones o duplicados ilegales que sobre sus programas se practique.





STRONTIUM DOG

Johny Alpha es un agente investigador destructor, un cazador de recompensas del futuro. Johny lleva consigo electrobengalas, que deslumbrarán a sus enemigos, obligándoles a dejar de disparar, y bombas de tiempo. Buena suerte en tu misión.

BROAD STREET

... 7 Personajes, 10 acordes perdidos, 15 horas, 48 guardias de tráfico, 95 estaciones de metro, 45.000 m^2 de Londres, 7 millones de londinenses, 943 pantallas con trepidantes acciones.

ASTRO BLASTER

5 Olas de ataque, 15 niveles de dificultad, fuego rápido, tormenta de meteoritos, alienígenas mutantes, joystick, bombas asesinas, gráficos a todo color en alta resolución, sonido, naves volantes, bolas de plasma, rayos laser.

PURPLE TURTLES

Basado en las legendarias tortugas púrpuras. Un juego para los jóvenes y demás personas que se hayan cansado de las palizas de los marcianitos.

Recorte y envíe este cupón a POWER, C/. Nápoles, 98, 1.° - 08013 BARCELONA, o llame al teléfono (93) 232 24 61

Deseo recibir los juegos que a continuación especifico, comprometiéndome al pago del importe de los mismos.

Nombre ____________

Dirección ____________

Teléfono ____________

Firma:

☐ Contrarreembolso. ☐ Adjunto Talón. ☐ Giro Postal.

Deseo recibir información de sus programas en: MSX ☐ AMSTRAD ☐

TITULO	SISTEMA	PRECIO UNIT.	CANTIDAD
LASER ZONE	Spectrum	975 Pts.	
GRIDRUNNER	Spectrum	975 Pts.	
FRENZY	Spectrum	1.275 Pts.	
ASTRO BLASTER	Spectrum	1.275 Pts.	
QUINTIC WARRIOR	Commodore	1.275 Pts.	
PURPLE TURTLES	Commodore	1.275 Pts.	
STRONTIUM DOG	Spectrum	1.800 Pts.	
STRONTIUM DOG	Commodore	2.100 Pts.	
FRIDGE FRENZY	Spectrum	1.800 Pts.	
BROAD STREET	Spectrum	2.400 Pts.	
BROAD STREET	Commodore	2.600 Pts.	
EVIL CROWN	Spectrum	2.400 Pts.	
EVIL CROWN	Commodore	2.600 Pts.	

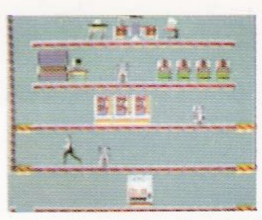
IMPOSSIBLE MISSION™

YOUR MISSION–TO SAVE THE WORLD.

EPYX™ COMPUTER SOFTWARE

Strategy Games for the Action-Game Player

TOP SECRET



Como miembro del Escuadrón Antiterrorista de Computadores (EAC), su misión es encontrar y coger al infame Elvin, quien tiene a la población mundial como rehén bajo la amenaza de aniquilación nuclear. Debe pasar por un pasillo a través de habitaciones y túneles en su cuartel general tratando de evitar los robots protectores de Elvin.

Deberá correr más o saltar por encima del siguiente robot o jugar seguro y tomar el tiempo necesario para unir los códigos para desactivar los robots, encontrar a Elvin y pararle.

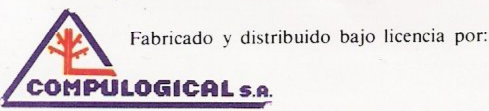
P. V. P.: COMMODORE 64, 2.300 ptas.
P. V. P.: SPECTRUM 48 K, 2.300 ptas.

Utilice su cámara para fotografiar tantas pistas como sea posible para encontrar la contraseña que le permitirá penetrar en la habitación de control de Elvin.

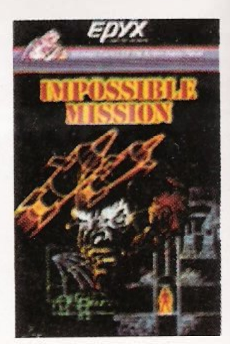
Su misión: Salvar al mundo, ¡pero deprisa!

Un jugador; control joystick o teclado (sólo SPECTRUM).

Fabricado y distribuido bajo licencia por:



Santa Cruz de Marcenado, 31 - 28015 Madrid - Teléf. 241 10 63



Distribuido en CATALUÑA y BALEARES por: DISCLUB, S. A.
Balmes, 58 08007 BARCELONA Teléfono (93) 302 39 08



DIRECTOR:
Alejandro Diges
COORDINADOR EDITORIAL:
Francisco de Molina
DISEÑO GRAFICO:
Tomás López
COLABORADORES:
Antonio Taratiel, Luis R. Palencia, Marchella Pellegrini, Francisco Tórtola, José Pérez Montero, Benito Román
INPUT Commodore es una publicación juvenil de EDICIONES FORUM

GERENTE DIVISION DE REVISTAS:
Angel Sabat

PUBLICIDAD: Grupo Jota
Madrid: c/ Cuenca, 1, 1.º
Teléf. 253 45 01/02
Barcelona: Avda. de Sarriá, 11-13, 1.º
Teléf. 250 23 99
FOTOMECANICA: Ochoa, S. A.
COMPOSICION: EFCA, S. A.
IMPRESION: Edime, S. A.
Depósito legal: M. 27.884-1985
Suscripciones: EDISA,
López de Hoyos, 141. 28002 Madrid
Teléf. (91) 415 97 12
Redacción:
Alberto Alcocer, 46, 4.º
28016 Madrid. Teléf. 250 10 00
DISTRIBUIDORA:
R.B.A. PROMOTORA DE EDICIONES, S. A.
Travesera de Gracia, 56. Edificio Odiseus.
08006 Barcelona
Se solicitará el control OJD

INPUT commodore

Sumario

AÑO 1 NUMERO 3

LOS NUEVOS 128

Estamos en el centro de la época más activa del año para el sector informático. A finales del pasado mes tuvo lugar el **Sonimag**, certamen en el que los microordenadores toman cada vez un mayor protagonismo.

En ese mismo lugar fue presentado en sociedad el **Spectrum 128** y también se pudo ver el **Commodore 128.** Todo el mundo anda expectante con los 128 Kbytes de RAM, pero tal vez el brillo esconda la verdadera realidad y los nuevos modelos no tengan una aceptación tan rápida como pudiera suponerse. De todas formas, al usuario le gusta saber que su ordenador está soportado por otro modelo con mayor memoria y, sobre todo, compatible en *software* con el anterior.

Pero tal vez estos modelos solamente sean un puente para los inquietos no dispuestos a esperar tranquilamente la llegada de la nueva generación de máquinas con un mayor coeficiente de lo que los estadounidenses han dado en llamar *«user friendlyness»,* ordenadores de manejo más interactivo y sencillo por parte del usuario, como son los **Amiga,** de **Commodore,** y el **520** de **Atari.**

El evento que sigue es en Madrid esta vez, el **SIMO,** a finales del presente mes. Es la puerta hacia los treinta días en los que se vende del cuarenta al cincuenta por ciento de los microordenadores previstos para el año.

El próximo mes, **INPUT** ofrece un extra, consistente en un cuadernillo de programas listos para ser copiados y disfrutados.

En algunos programas hemos hecho la experiencia piloto de sustituir los espacios y caracteres entrecomillados por su equivalente entre corchetes. Si el método os parece más cómodo no dudéis en hacérnoslo saber.

LOS MEJORES DE INPUT

Hemos pensado que es interesante disponer de un ***ranking*** que ponga en claro, mes a mes, cuáles son los programas preferidos de nuestros lectores. Para ello, es obligado preguntaros directamente y tener así el mejor termómetro para conocer vuestras preferencias. Podéis votar por cualquier programa aunque no haya sido comentado todavía en **INPUT**.

El resultado de las votaciones será publicado en cada número de **INPUT**.

Entre los votantes sortearemos 10 cintas de los títulos que pidáis en vuestros cupones.

Nota: No es preciso que cortéis la revista, una copia hecha a máquina o una simple fotocopia sirven.

Enviad vuestros votos a: **LOS MEJORES DE INPUT** Alberto Alcocer, 46 - 4.° B. 28016 Madrid

ELIGE TUS PROGRAMAS

Primer título elegido ____________________

Segundo título elegido ____________________

Tercer título elegido ____________________

Programa que te gustaría conseguir ____________________

Qué ordenador tienes ____________________

Nombre ____________________

1.er Apellido ____________________

2.° Apellido ____________________

Fecha de nacimiento ____________________

Teléfono ____________________

Dirección ____________________

Localidad ____________________

Provincia ____________________

INPUT COMMODORE N.° 3

SCHIZOFRENIA

or how to be in two minds about paranoia

QUICKSILVA

CBM 64

ESTE PROGRAMA
APARECE
SIMULTANEAMENTE
EN INGLATERRA
Y ESPAÑA

OTROS TITULOS

GAMES DESIGNER MSX
ANT ATTACK MSX
ANT ATTACK CBM 64
ANT ATTACK SPECTRUM 48
SEE-SAW CBM 64
TRASHMAN CBM 64
GATECRASHER SPECTRUM 48
XADOM SPECTRUM 48
FALL OF ROME SPECTRUM 48
FALL OF ROME CBM 64



SCHIZOFRENIA
o el desdoblamiento
de personalidad

Alphonse T. Nurd necesita ayuda. Su otro YO ha descubierto el sentido de independencia. Si pueden descubrir los secretos de los separadores sub-atómicos de partículas y limpiar ascensores, éste es tu juego. El rompecabezas de una vida... o dos.

DSTRIBUIDO EN EXCLUSIVA EN ESPAÑA POR REDIS, S. A.
c/. Regas, 13. 08006 BARCELONA. Tels. 218 21 23 - 218 22 46

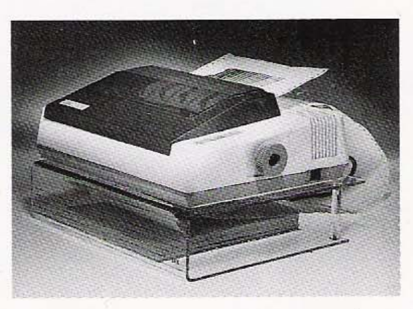


ACTUA LA JUSTICIA

A mediados del pasado mes, el diario El País daba la noticia. En Alicante, la policía acababa de desactivar una estructurada organización de copia de *software* a nivel industrial.
El juez ordenó el ingreso inmediato en prisión de los responsables de la fabricación-presuntamente ilegal.
Según fuentes bien informadas, esta operacion no sería más que la punta del *iceberg* de una operación mas basta, aún sin concluir.



PIE PARA IMPRESORA



Una ingeniosa idea resuelve el problema de espacio para el papel. La empresa First importa este soporte de Ian Corp.



GUIA DE SOFTWARE

La Generalitat de Catalunya, por medio del Centro Divulgador de la Informática, ha producido un catálogo-guía de 92 páginas, donde aparece una lista exhaustiva de programas de aplicación profesional, por sectores de utilidad y la referencia del proveedor.
Asimismo se reflejan los sistemas operativos bajo los que han sido desarrollados.
En el directorio de empresas aparece un dato interesante: la persona de contacto.



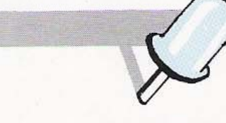
MADE IN SPAIN

Siguiendo la pauta que en su día iniciase Indescomp, la firma Idealogic ha decidido fabricar sus propios *joysticks* a un precio competitivo (1.990 pts.), por la ausencia de tasas arancelarias.
Construido integramente en plástico, el conector situado al otro extremo del cable es el clásico estándar "D", directamente enchufable a los ports de los Commodore.



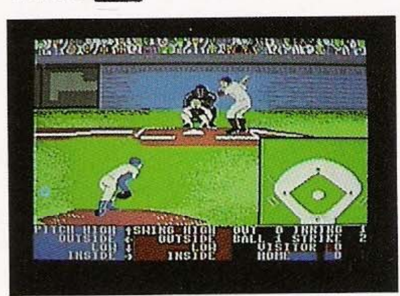
NUEVA ASOCIACION

Un grupo formado por nueve empresas catalanas de distribucion de *software* ha creado una asociacion profesional destinada a proteger los intereses propios, asi como los del consumidor, según reza su declaración de principios.
Las primeras firmas que figuran en la lista son: Ace Software, Companyia General de Software, Cimex Electrónica, Dimensión New, Fromesoft, Idealogic, Software Center, Scanner Software y Ventamatic.

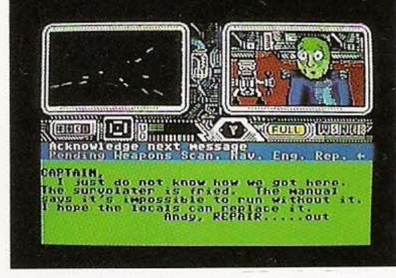


ACOLADE, UN PASO ADELANTE

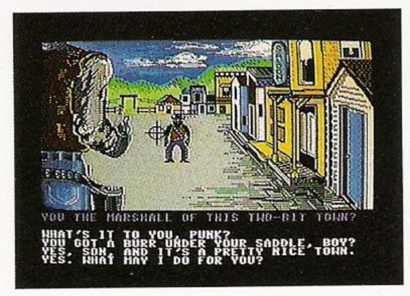
El Commodore 64 dispone de unas capacidades gráficas aún por descubrir. Son los programadores de élite quienes saben obtener el mayor provecho de ellas, en base a su experiencia acumulada. Este es el caso de la nueva firma amareciana Acolade, formada con programadores procedentes de la mítica Epyx.

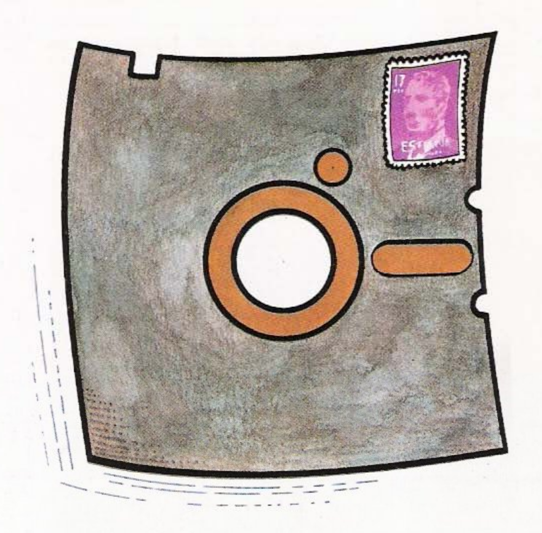


La calidad de los juegos que ha presentado es tan depurada que los personajes protagonistàs dan una sensación de realidad próxima a los dibujos animados.
El primer juego que han puesto en circulación tiene que ver directamente con la sociedad americana. Es un partido de *Baseball*, en el que cada jugador es controlado por los *joysticks*. Con uno se determina la posición del guante del



catcher asi como el lanzamiento de la pelota. El otro sirve para *batear*. l siguiente título que lanzarán se encuadra en las aventuras de naves espaciales. Un desafío en la calle principal de un poblado del lejano oeste es el tema elegido para completar la primera terna de programas.
La firma Compulogical será la encargada de distribuir este *software* en nuestro país.





EL BUZON DE INPUT

Por mucha información que hemos buscado, no hemos encontrado nada y lo que hay está muy mal explicado. Estos son los dos tipos de duda:

La primera es cómo imprimir *pixels* en la pantalla, o sea, que POKEs se deben utilizar para imprimirlos en diferentes puntos de la pantalla y de distintos colores, para así formar, por ejemplo, un dibujo hecho con *pixels*.

La segunda es cómo utilizar el *port* del usuario como entrada de señal, es decir, hacer que una señal entre en el *port* del usuario, enviando la señal al interior del ordenador por medio del *chip* CIA. Y también cómo se utiliza la salida de Audio/Video para introducir señal de audio en el 64.

J. M. Folch. Sabadell.

*El **64**, como todos los ordenadores, dispone de un área de memoria cuyas direcciones están directamente relacionadas con las posiciones de los distintos puntos* (pixels) *de la pantalla. Este ordenador tiene un formato de 200 por 320 puntos o, lo que es lo mismo, 64.000 puntos. Cada uno de ellos pueden ser visible o invisible, encargándose de ello un bit específico (de ahí el calificativo* bit-mapped*). La información depositada en la memoria se agrupa en forma de bytes (8 bits), por lo que la cantidad de memoria requerida en este caso es de 8000 bytes.*

La organización de la memoria utilizada por la pantalla queda para un futuro artículo, pero de momento podemos trabajar con la siguiente fórmula:

*Dirección a POKEar = 8192 + INT(Y/8)*320 + (Y1 AND 7) + INT(X/8)*8*

Siendo Y un número comprendido entre 0 y 24 (el número máximo de líneas de texto presentes en la pantalla). Y1 está comprendido entre 0 y 7 (la línea seleccionada dentro del bloque que la contiene) y X es un número entre 0 y 39 (número máximo de caracteres por línea). Una vez obtenida la dirección, tenemos acceso a 8 pixels *consecutivos. Si POKEamos un 1 se enciende el primer* pixel*, con 2 el segundo, un 4 ilumina al tercero, 8 el cuarto, 16 el quinto, 32 el sexto, 64 el séptimo y 128 el octavo. La suma de varias de estas cantidades activa simultáneamente a los* pixels *correspondientes.*

*Esta agrupación de 24 por 40 «super*pixels*» de 8 por 8* pixels *obedece a una estrecha relación entre las pantallas de texto y alta resolución.*

Cuando POKEamos una de estas direcciones no conseguimos ver nada, a menos que tecleemos:

POKE 53265, PEEK (53265) OR 32: POKE 53272, PEEK (53272) OR 8

que nos pone en modo gráfico de alta resolución.

El color es otra cuestión. Aquí no podemos dotar al pixel *de un color independiente de sus adyacentes. Vuelve a surgir la organización de 25 por 40. Cada una de las agrupaciones de 8 por 8 puede disponer de un color de primer plano y otro de fondo. El bit «iluminado» recibe el color de primer plano y el «apagado» el de fondo.*

Las direcciones de memoria comprendidas entre 1024 y 2023 almacenan la información del color. Todos estos bytes se pueden dividir en dos (nybbles)*. Los cuatro bits de menor peso (0 al 4) definen el color de fondo con 16 posibilidades distintas. Los restantes hacen lo mismo, pero con el color de primer plano.*

La Entrada/Salida de señales a través del chip CIA proporciona material para escribir un nutrido artículo, el cual no rechazamos preparar.

En cuanto a la entrada de señal por el conector de Audio/Video hacia el chip *SID es factible, pero hay que manejarlo con extremo cuidado, pues podríamos dañar al ordenador. Igualmente, la manipulación de esta señal es el tema de un artículo que estamos elaborando.*

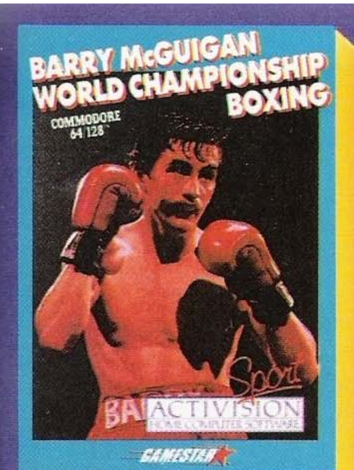
El programa «Así lo oye tu ordenador», de la pág. 12 del N.º 1, no me funciona. Una vez puesto el Simon's BASIC **y tecleando el listado, pongo RUN y el ordenador me contesta con un SINTAX ERROR en la línea 30. Si echáis un vistazo a la línea en cuestión veréis que es SYS 832. Decidme qué debo hacer, pues estoy desconcertado.**

A. A. Blanes.

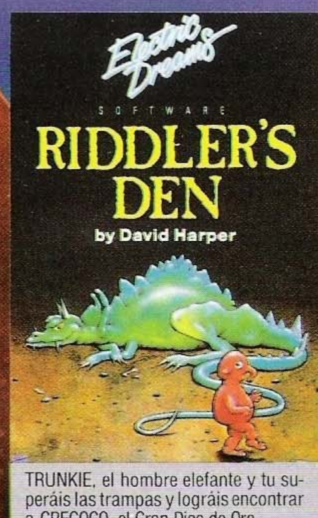
En principio, no existe ninguna razón aparente para que SYS 832 devuelva un mensaje de error de sintaxis. Este comando hace que comience a ejecutarse una rutina en código máquina, que en este caso comienza en la dirección de memoria 832, a partir de la cual hemos cargado algo de nuestro programa, empleando el bucle con READ, POKE y los DATAs.

El programa, vuelto a repasar, no presenta errata alguna. Intenta hacer lo siguiente: Borra la línea 30, ejecuta el resto del programa y escribe directamente SYS 832.

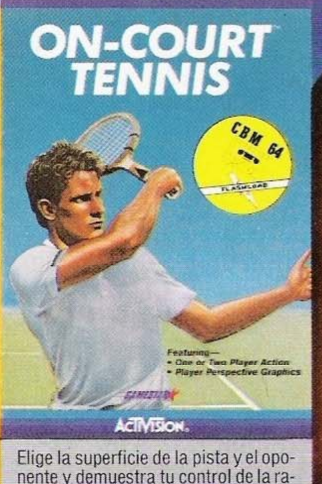
*Por otro lado, comprueba que has seguido la secuencia de trabajo correcta. Primero has de enchufar el cartucho del **Simon's BASIC,** conectando después la alimentación al ordenador. A continuación teclea el programa y ejecútalo. Si por cualquier circunstancia no hubieras puesto previamente el cartucho, es normal que aparezcan mensajes de error. Igualmente, si hubieras cargado el programa en cinta o disco y lo vuelves a cargar en el ordenador sin el cartucho, vuelven a surgir los problemas de sintaxis.*



Juego en el que puedes crear a tu propio boxeador. Elige su raza, estilo físico e imagen. Entrénale y demuestra sus habilidades. C.S.A.



TRUNKIE, el hombre elefante y tu superáis las trampas y lográis encontrar a GREGOGO, el Gran Dios de Oro. C.S.A.



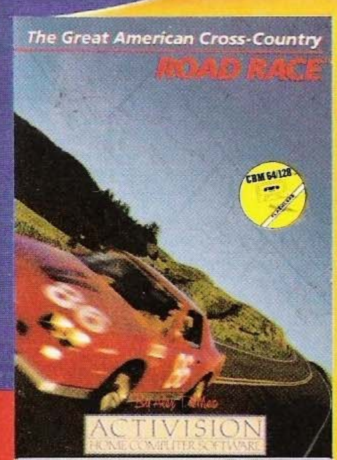
Elige la superficie de la pista y el oponente y demuestra tu control de la raqueta jugando a dobles o individual. C.S.A.



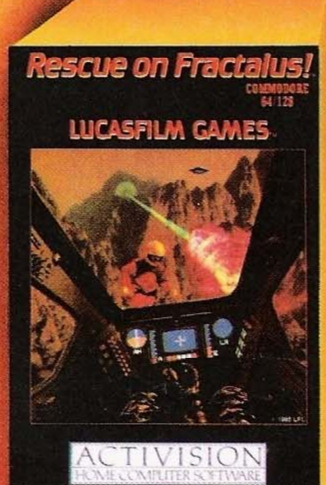
Entra en el mundo ciclista a través de las 16 etapas del Tour. Con acompañamiento musical y el jersey amarillo esperando al ganador. C.

ACTIVISION INC
HOME COMPUTER SOFTWARE

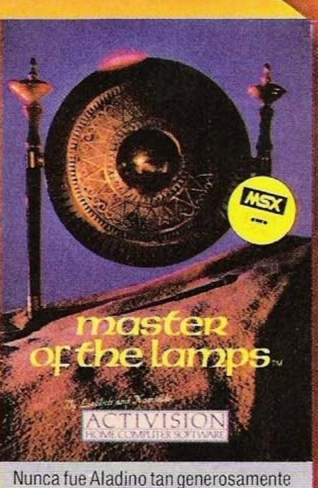




Al volante de tu coche atraviesas 25 ciudades. Seleccionas la ruta, maniobras a través del tráfico... Todo un reto de conducción automovilística. C.A.



Recorre un planeta en tres dimensiones a la búsqueda de tus compañeros, mientras el enemigo te persigue. C.S.A.

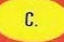


Nunca fue Aladino tan generosamente premiado por los genios.
Vuele sobre una increíble alfombra mágica en tres dimensiones. C.A.M.

HACKER
You've found your way in. But is there a way out?
COMMODORE CBM 64/128
ACTIVISION
HOME COMPUTER SOFTWARE
By Steve Cartwright

Compleja aventura donde los jugadores deben buscar a través de las diferentes pistas y problemas como resolver el misterio. C.S.A.

... y sus clásicos: GHOSTBUSTER. C.S.A.M. DECATHLON DE ACTIVISION. C.M. RIVER RAID. C.S.M.

* ENDURO S PITFALL 2 C.S.M. y otros títulos más a 1.540.- P.V.P. • SPACE SHUTTLE. C. DESIGNER PENCIL.  ... etc.

Disponibles para:
COMMODORE C
SPECTRUM S
AMSTRAD A
MSX M

EN TIENDAS ESPECIALIZADAS Y GRANDES ALMACENES, O DIRECTAMENTE POR CORREO O TELEFONO A:

Distribuido por:
PROEIN, S.A.
Velázquez, 10 - 28001 Madrid
Tels. 91/276 22 08/09

EL BUZON DE INPUT

En un artículo de la revista número 1, «Cuidado de cintas y *diskettes*», dicen que es conveniente hacer copias de seguridad por imprevistos accidentes. Sin querer entrar en la polémica, me pregunto si es ser «pirata» el intentar hacer una copia de un programa comercial, porque no debe de hacer mucha gracia el gastarse una cierta cantidad de dinero (a veces bastante) y que por un accidente se te fastidie el programa, con la consiguiente pérdida de dinero. Otra cosa muy diferente es hacer negocio con ello, pero piensen que un programa pirateado te cuesta un 10% (diez) del valor real.

Mi intención no es criticar esto, ni pedir que bajen los precios, sino una solución para que no me vuelva a pasar otro accidente..., tampoco he querido defender a los piratas, aunque no niego que es sugestivo tener el mejor juego por muy poco dinero.

Javier Casado. Móstoles (Madrid)

No es nuestra misión determinar si se le puede adjudicar a uno el calificativo de pirata por hacer tal tipo de copias, pero desde luego con toda seguridad podemos afirmar que se trata de algo ilegal, por cuanto se dañan los intereses de autores, firmas comercializadoras, canales de distribución, etc.

Estamos de acuerdo con que es una triste gracia que en un momento dado se nos deteriore un programa. Ahora bien, si procede de un distribuidor serio, éste no pondrá el menor impedimento para facilitarnos una copia nueva a la entrega de la defectuosa, bien directamente o (en algunos casos) a través del comercio que lo vendió. Esta afirmación es real, puesto que hemos tomado la precaución de contrastarla con algunos importadores españoles. Al fin y al cabo, cuando se compra un programa legalmente es lo menos que se puede pedir. Esto es sin duda algo que no puede garantizarse con los programas de procedencia dudosa.

Hacer copias ilegales puede terminar siendo perjudicial para todos. Veamos cómo.

Indudablemente la aparición de nuevos títulos, cada vez mejores y más sugestivos, tiene una importante motivación de índole económica. En muchos casos el autor suele ser alguien tan anónimo como cualquiera de nosotros, quien después de pasar cantidad de horas frente al teclado decide vender a una casa de software *el fruto de su imaginación. En otros casos el programa es desarrollado por el personal de plantilla de estas firmas. La aparición de copias ilegales merma en gran manera la creación de nuevos títulos, debido a la minimización del aliciente económico.*

En gran medida, ésta es una de las principales razones para que no se desarrolle software *de alta calidad en nuestro país. ¿Cuántos de nuestros lectores estaríais dispuestos a trabajar durante meses en un programa del que después se venderían solamente unas decenas de unidades, apareciendo a renglón seguido miles de copias ilegales? Afortunadamente para nosotros, el mercado anglosajón tiene mayores dimensiones y menos «piratas» o, al menos, ellos siguen apostando (por el momento).*

CLUB de SOFT de ESPAÑA
CBM 64

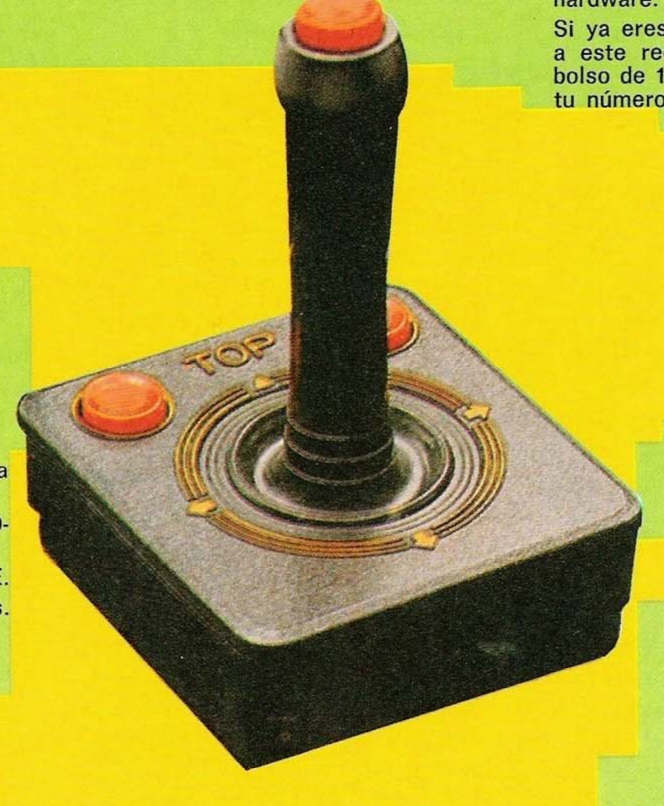
¿COMO PUEDES HACERTE SOCIO DEL CLUB DE SOFT DE ESPAÑA?

— Sólo tendrás que abonar 1.000 ptas. como cuota UNICA de inscripción.
— Como regalo de Bienvenida a nuestro Club de Soft de España, recibirás completamente gratis UN MARAVILLOSO JOYSTICK.
— Pertenecer al Club no te obliga nunca a comprar ni pagar cuotas posteriores.

¡Hazte socio del Club de Soft de España! Tendrás multitud de ventajas...

— Recibirás puntual información de las novedades de soft que haya en el mercado. Te enviaremos un catálogo bimensual de programas para Commodore 64.
— Podrás adquirir en el Club, programas de nuestros catálogos, con un 30 % de descuento sobre su precio en el mercado.
— Obtendrás descuentos de hasta un 20 % sobre hardware.

Si ya eres socio del Club y deseas acogerte a este regalo, puedes hacerlo contra reembolso de 1.000 ptas., pero no olvides decirnos tu número de socio.



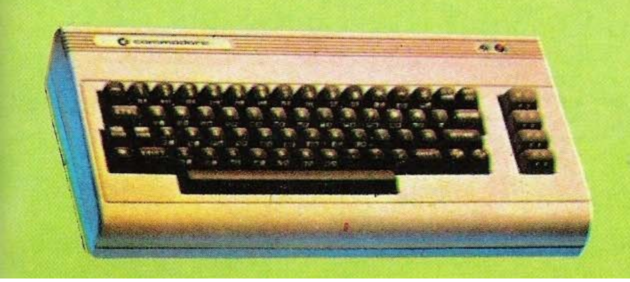


☆ 3 botones de disparo, dos en la base y uno en la palanca.
☆ Conexión directa con COMMODORE. No necesita interface.
☆ RESISTENTE, FLEXIBLE, SENSIBLE.
☆ Precio en el mercado: 1.900 ptas.

Más de 5.000 socios en toda España, ya se benefician de las ventajas de nuestro CLUB.

DIFUSION POR AMISTAD

Habla a tus amigos de las enormes ventajas que supone pertenecer al CLUB DE SOFT DE ESPAÑA. Si junto con tu inscripción se inscriben **dos** amigos tuyos por tu recomendación, te **regalamos** un programa de juegos, y ellos recibirán igualmente el joystick como regalo de bienvenida.



CUPON DE INSCRIPCION AL CLUB DE SOFT DE ESPAÑA

Apdo. 52086 - Eraso, 13 - Tel. 246 10 94 - 28028 MADRID

Nombre ____________________
Dirección ____________________
Ciudad ____________ C. P. ________ Tel. ________
Deseo pertenecer al CLUB DE SOFT DE ESPAÑA. Ruego me envíen como regalo de bienvenida al CLUB el programa
N.º de socio Firma

El CLUB DE SOFT DE ESPAÑA te enviará el programa solicitado, el joystick, junto con un reembolso de 1.000 ptas. más gastos de envío, como cuota de inscripción al Club.

DIFUSION POR AMISTAD

NOMBRE de un amigo ____________________
Dirección ____________________
Ciudad ____________ C. P. ________ Tel. ________
Programa de regalo de bienvenida
Firma

DIFUSION POR AMISTAD

NOMBRE de un amigo ____________________
Dirección ____________________
Ciudad ____________ C. P. ________ Tel. ________
Programa de regalo de bienvenida
Firma

BUSCANDO EL MEJOR CAMINO

- EL CAMINO CRITICO
- AHORRO DE TIEMPO
- LA PLANIFICACION DE UN PROYECTO
- SUCESOS Y ACTIVIDADES

Si estás intentando organizar un proyecto de cualquier clase, ya se trate del mantenimiento de tu coche o de comprarte una casa, este programa te ayudará a planificarlo y a ahorrar tiempo.

¿Alguna vez te has encontrado parado a la mitad de un trabajo, como, por ejemplo, cuando decoraban tu cuarto de estar y te has dado cuenta de repente que si hubieras planeado las cosas adecuadamente no tendrías que sentarte a esperar que se seque el yeso antes de empezar a pintar? O a lo mejor al arreglar tu coche te has encontrado que si lo hubieras pensado bien antes, habrías tomado la precaución de sustituir el interruptor de la luz de los frenos antes de poner el cilindro bomba, con lo cual ahora no tendrías que volver a extraerlo.

Cualquier actividad compleja necesita de una cierta planificación, pero a veces parece ser necesario hacer tantas cosas, todas con una duración distinta, y dependiendo además de que las actividades anteriores se hayan completado con éxito, que resulta casi imposible hacer lo adecuado en el momento correcto. Lo normal es que a toda actividad le corresponda un instante de tiempo muy bien definido en el que ha de terminar.

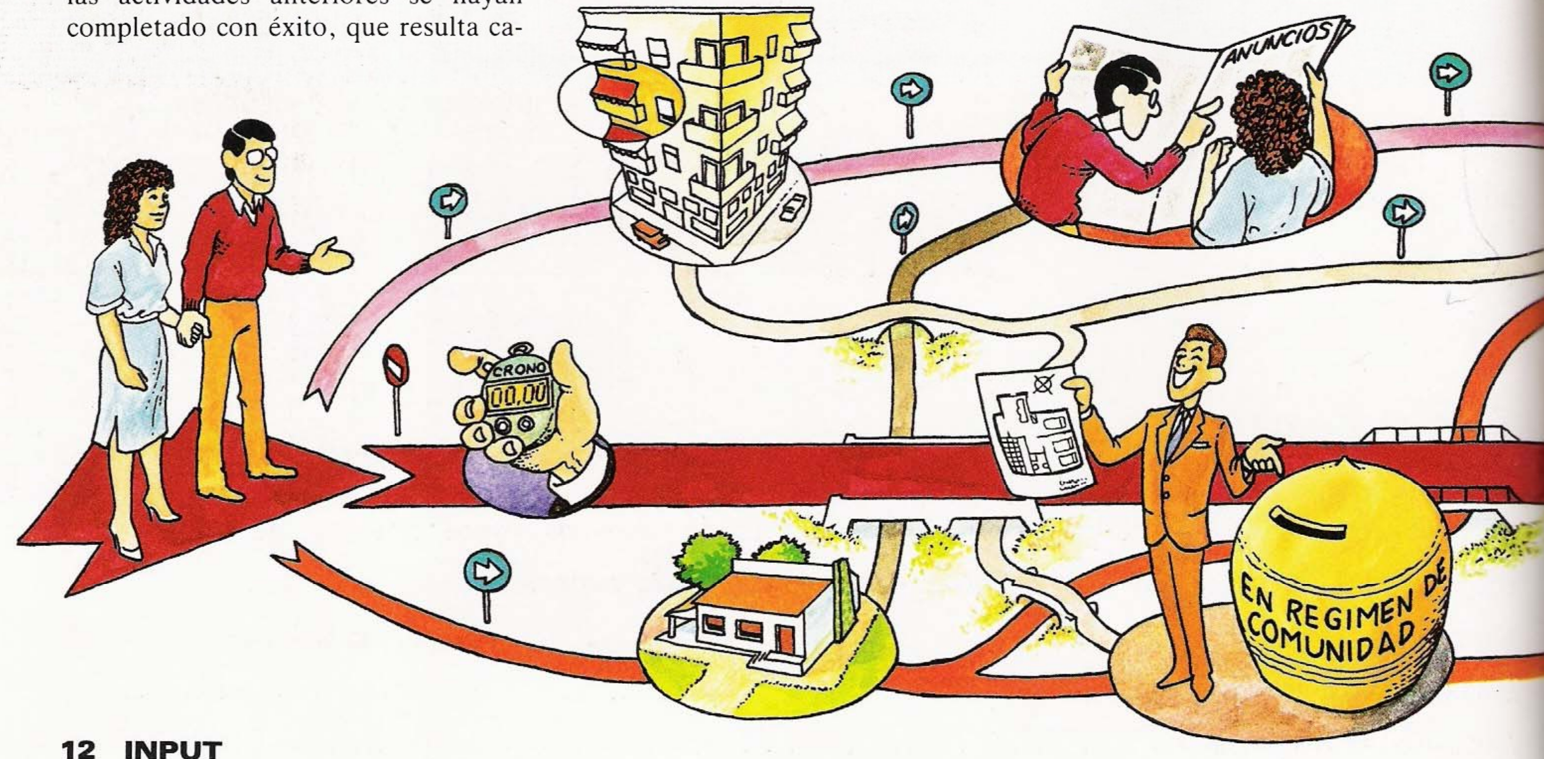
EL CAMINO CRITICO

Si examinas el tiempo requerido por cada actividad, encontrarás siempre que hay una cadena de actividades que determina el tiempo total requerido por el proyecto completo. A esta cadena de sucesos se le llama **camino crítico,** y cualquier retraso o adelanto en esta cadena alterará la fecha final de terminación del proyecto. Es posible que existan otras actividades en las que un retraso no afecte a la duración total del proyecto. El desarrollo de estas ideas no es tarea fácil si tienes que dibujar a mano el camino crítico. Pero con ayuda de tu ordenador y del programa que damos a continuación, el análisis de la mejor situación posible resulta mucho más sencillo. El programa te permite construir una base de datos que contiene todas las actividades que requiere un programa particular, así como la duración de las mismas (o un valor estimado si no se conocen) y el orden en que han de ser ejecutadas. A continuación calcula el camino crítico y te da la holgura temporal disponible en las actividades no críticas, de forma que sepas si tienes alguna tardanza y hasta qué punto te puedes demorar en cada una sin extender la duración total del proyecto.

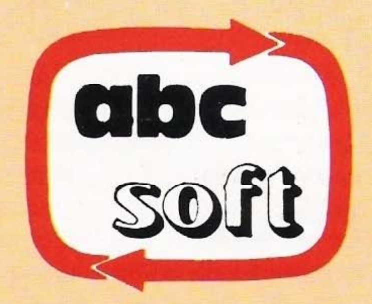
El programa tiene dos partes. La primera, que es la que acompaña a este artículo, establece una base de datos de todas las actividades, mientras que la segunda, que veremos la próxima vez, calcula el camino crítico para que puedas decidir la forma más rápida y eficiente de realizar tu proyecto.

AHORRO DE TIEMPO

Como permite obtener un camino crítico, esta técnica de programación



SUPERNOVEDADES



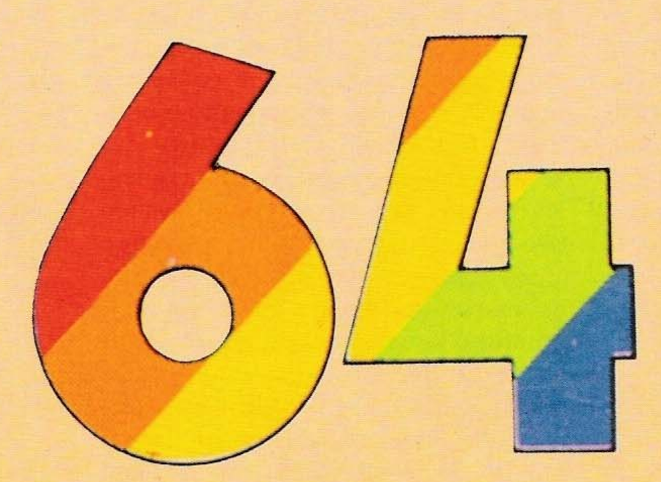
Santa Cruz de Marcenado, 31
28015 MADRID
Teléfs. (91) 248 82 13
(91) 242 50 59

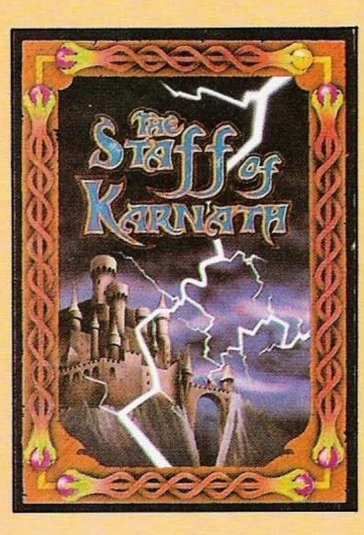
La emoción de la velocidad te la trae Digital Integration en Speed King. Fabulosa carrera de motos con posibilidad de elegir circuito, motocicleta de seis marchas...

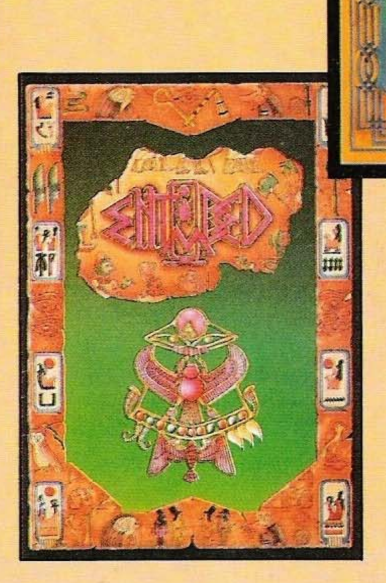
SPEED KING * C64
2.600 pts.

Fantástico simulador de vuelo con despegue vertical desde un portaviones, con el que podrá mantener una batalla con los aviones enemigos, hacer maniobras, ensayar vuelo...

JUMP JET * C64 y AMSTRAD
2.895 pts.







Diviértete con ULTIMATE y las aventuras de Sir Arthur Pendragon en The Staff of Karnath, Entombed y con la supernovedad BLACKWYCHE.

Si deseas recibir más información y propaganda de nuestros programas y de nuestras interesantes ofertas, por favor, rellena y envíanos este cupón.

Nombre .... Edad ....

Dirección ....

Localidad-Provincia ....

Ordenador .... Programas favoritos ....

se llama a veces Método del Camino Crítico o CPM *(Critical Path Method)* o Análisis del Camino Crítico, CPA *(Critical Path Analysis)*. Su otro nombre es PERT (Program Evaluation and Review Technique), que hace referencia al método de planificación por medio de redes. Estas técnicas se desarrollaron al principio de la década de los sesenta, siendo utilizadas en los programas espaciales de la **NASA.** Un ejemplo muy conocido fue el ahorro de tiempo que se obtuvo durante el programa de misiles **Polaris,** en los que el tiempo real transcurrido para obtener determinados resultados se redujo en un factor de cinco. Actualmente, los programas PERT se utilizan ampliamente en los negocios y la industria, para aumentar la eficacia en todo tipo de proyectos. Por eso, si tienes un pequeño negocio (o incluso uno grande) este, programa te resultará ideal.

PLANIFICACION DE UN PROYECTO

Pero el uso del PERT no está limitado al mundo de los negocios; cualquier proyecto, por pequeño que sea, puede ser evaluado con este programa. Realmente, el programa es útil, aunque estés haciendo algo más complicado que pintar una habitación o arreglar un coche. Un proyecto que mucha gente ha de realizar es la compra o venta de una casa, lo que supone coordinar muchas cosas diferentes, con frecuencia con una fecha límite muy estricta. La figura muestra un posible diagrama PERT para la compra de una casa. Los círculos encierran los llamados **sucesos.** Son los estados que hay entre las diversas actividades y en sí mismos no consumen tiempo. Delimitan el principio o el final de una actividad. Las actividades se señalan junto a las líneas que unen los sucesos, así como una estimación del tiempo requerido por cada una.

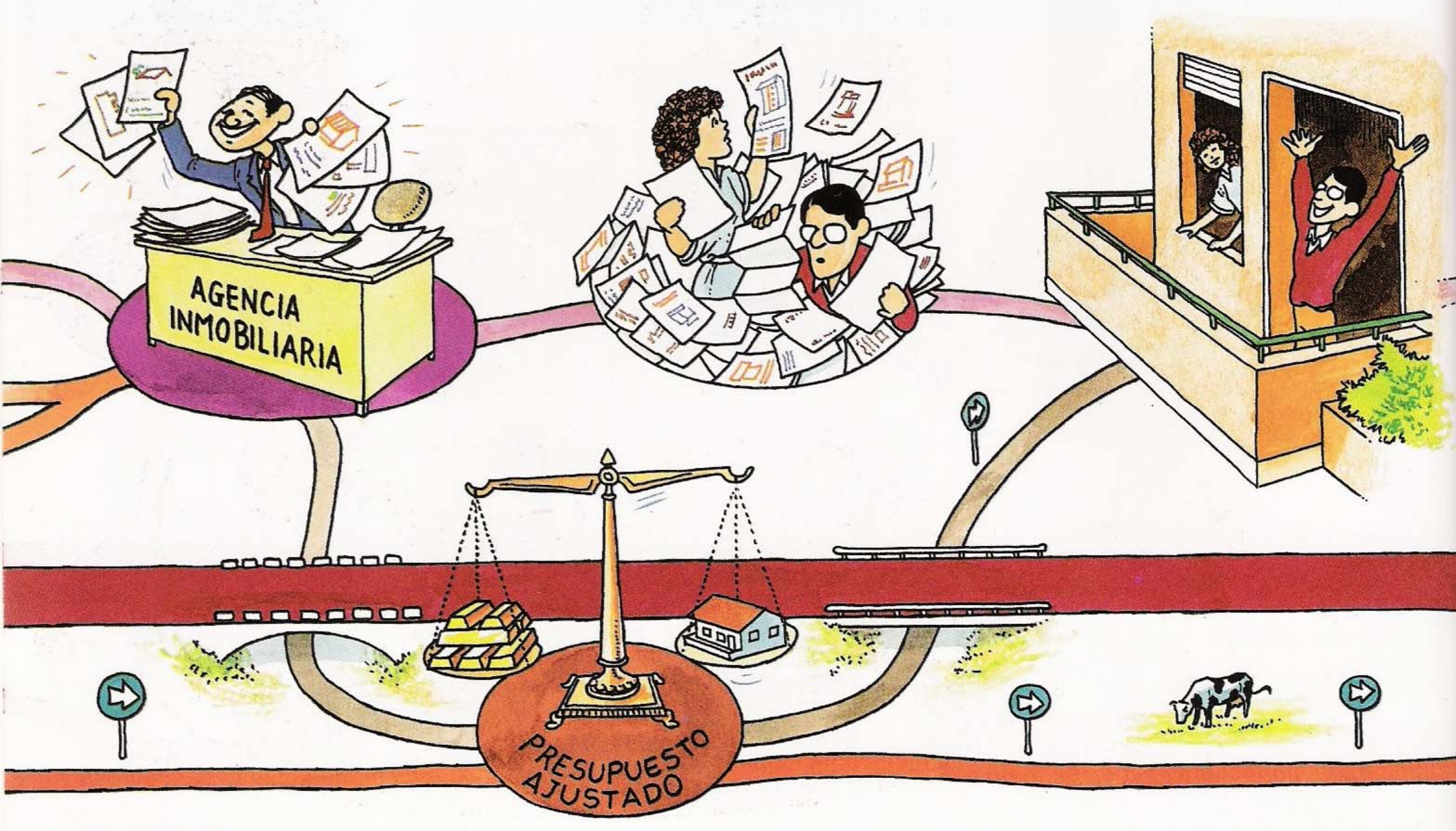
En el estado actual, muchas actividades son dudosas, y además habrás notado que hay muchos cruzamientos de líneas. Por eso, aunque toda la información esté allí, es difícil ver exactamente cuáles son las cosas importantes que hay que hacer rápidamente y si resulta posible o no hacer todo el trabajo en el tiempo permitido.

Las ventajas de utilizar este programa son enormes. Dispondrás de un listado claro de todas las actividades, que, además, podrás actualizar en cualquier momento. Entre toda la masa de información podrás identificar el camino crítico y controlarlo de cerca, pudiendo ejercer las acciones oportunas en cuanto se produzcan retrasos. También puede ser que descubras que hace falta replantear algunas actividades.

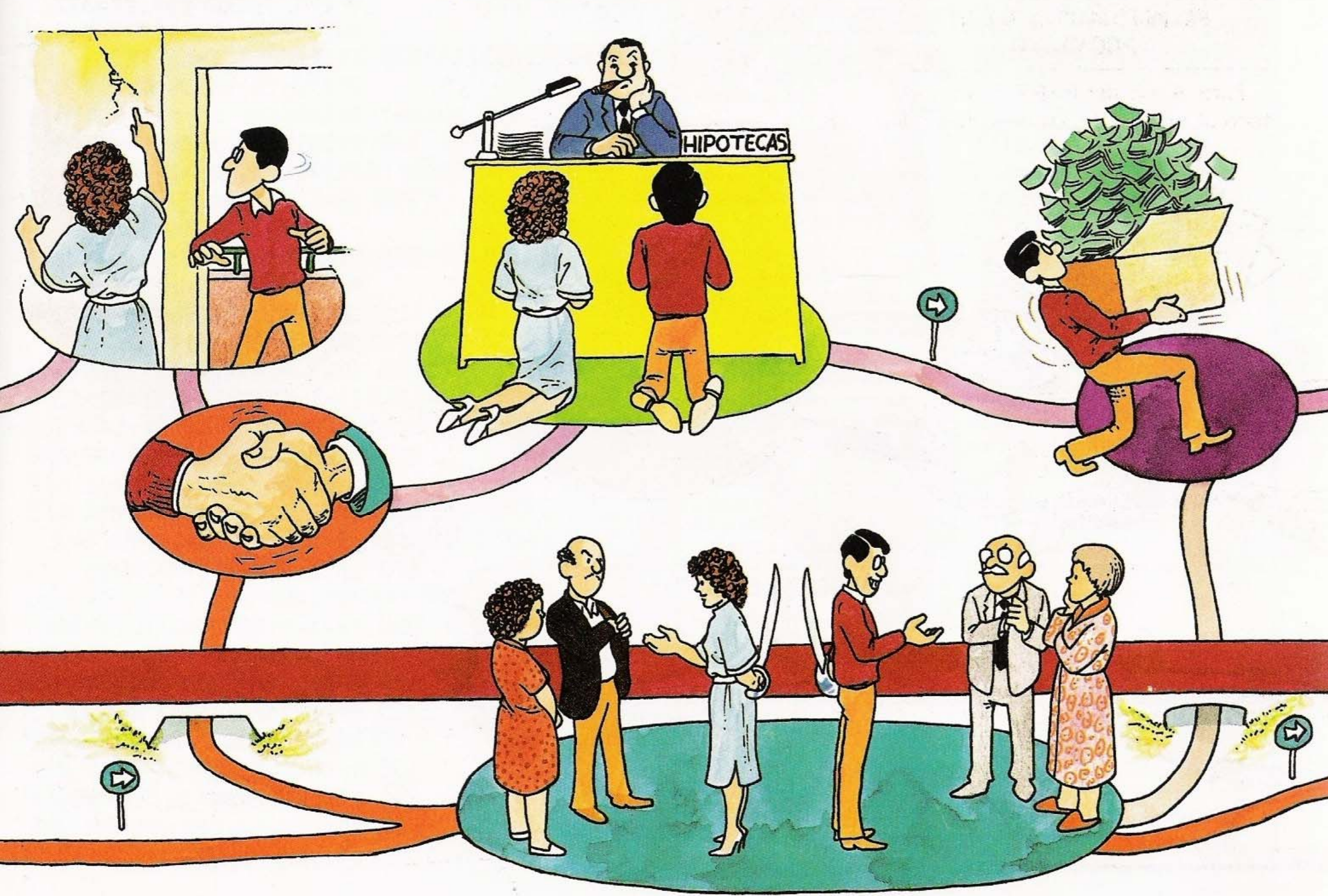
La próxima vez veremos cómo se introduce la información en el programa. Mientras tanto, teclea la primera parte y guárdala, dejándola lista para añadirle la segunda parte el mes que viene.

Teclea para Commodore-64

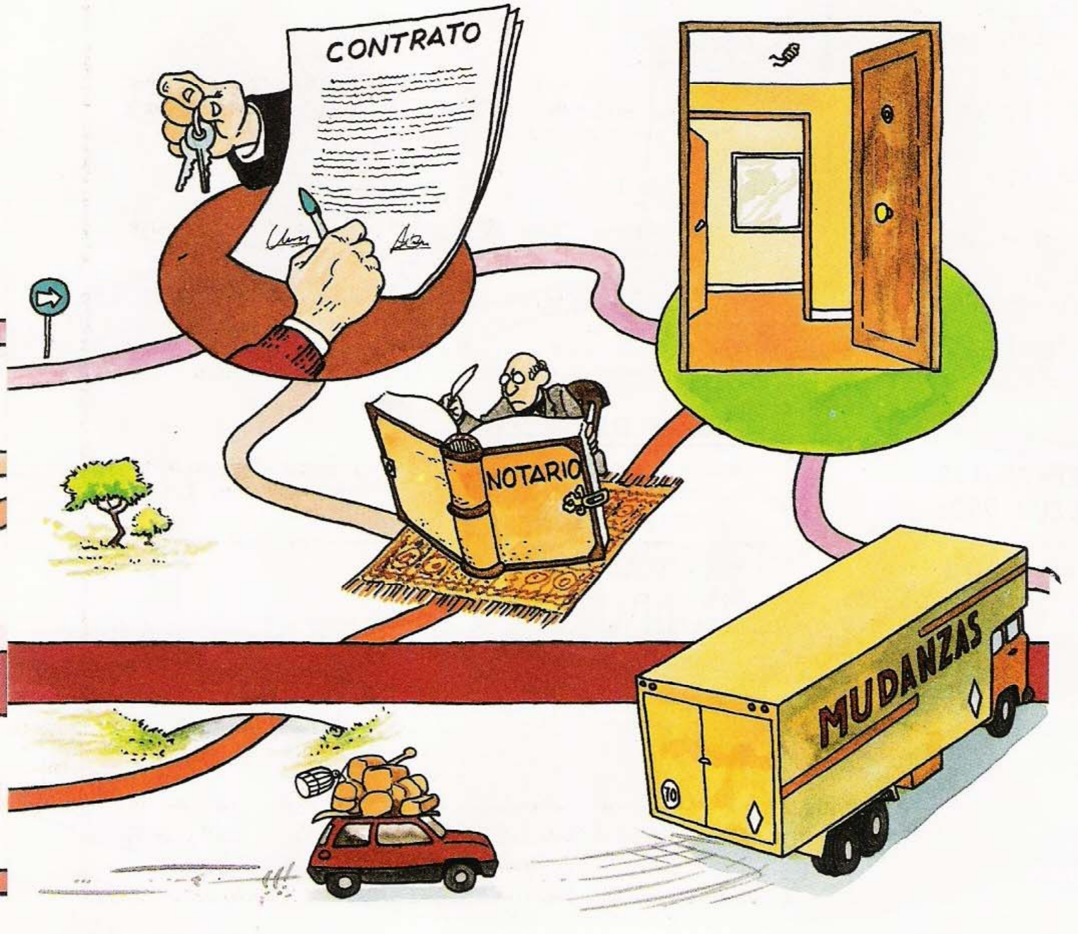
```
5 POKE 53280,1 :POKE53281,1
10 FA=0:MH=212:GOSUB12:CK=
   FA:GOTO50
12 CLS$="♥":CM$=",":ZZ=32766
   :MA=100:ME=100:QT$=CHR$
```



```
   (34):TR=-1:PRINTCL$
14 P$="ESCRIBE□":A$=
   "ACTIVIDAD"
16 DIMW$(5):W$(1)="NO□"+A$+
   □"SUCESO PRECEDENTE":W$
   (2)="NO PUEDES USAR TANTO"
18 W$(3)="NO PUEDES EMPLEAR
   ESTE NUMERO":W$(4)=P$+
   " TEXTO PARA ESTA□"
20 W$(5)=A$+"□ SE REFIERE A
   UN SUCESO NO DEFINIDO"
22 DEFFNA(X)=-X*(X<0):DEF
   FNZ(X)=-X*(X>0)
28 DIMA%(MA),G%(MA),U%(256)
30 DIMW(MA):DEFFNW(X)=-ABS
   (X)*(X<=1)-ABS(2-X)*(X>1)
32 DEFFNX(X)=X*(2.37572+X*X*
   (15.9402-X*X*(184.744-
   X*X*688.472)))/1.20667
40 DIM E%(ME)
44 DIM S%(MH),F%(MH),V%(MH),
   T(MH),N(MH),U$(MH),Y(MH),
   Z(MH)
46 DIM P(MH),Q(MH)
48 DEFFNU(XX)=TR*U%(ABS(XX)
   -(XX=0))*(XX>0):RETURN
   :LIST:
50 PRINT CLS$TAB(10)
   "       MENU       "
53 PRINT"■1=DEFINE■"A$:
   PRINT"■2=BORRAR■";A$
55 PRINT"■3=DEFINE SUCESO":
   PRINT"■4=BORRA SUCESO"
60 PRINT"■5=GUARDA LOS
   DATOS":PRINT"■6=CARGA
   LOS DATOS"
61 PRINT"■7=BORRA FICHEROS
   DEL DISCO":PRINT"■8=
   MUESTRA DETALLES"
62 PRINT"■9=REEMPEZAR"
63 PRINT"10=COMPRUEBA Y
   ORDENA LA RED"
64 PRINT"11=CALCULAR LOS
   TIEMPOS MEDIOS"
65 PRINT"12=CALCULAR CON
   INCERTIDUMBRE":PRINT"13=
   SALIDA A□";
66 IF KK$="S" THEN PRINT
   "■IMPRESORA■/PANTALLA"
67 IF KK$<>"S"THEN PRINT
   "IMPRESORA/■PANTALLA■"
68 PRINT"14=SALIDA PROGRAMA"
69 T=0:INPUT"■■■■ QUE
   OPCION";T
70 IF T=14THEN INPUT "ESTAS
   SEGURO(S/N)";AN$:IF AN$=
   "S"THEN SYS 58648:END
71 IF T=13 OR T=14 THEN 100
72 IF T=9 THEN 900
73 IFT<1 OR T>13 THEN PRINT
   "■■■■ OPCION"□;T;"NO
   ENTENDIDA":GOTO 114
74 IF T>10 AND NOT(CK) THEN
   PRINT"■■■■ COMPRUEBA
   PRIMERO LOS DATOS":
   GOTO 114
76 IF AA=0 AND(T>7 OR T=5)
   THEN PRINT"■■■■ NO
   PUEDO-NO HAY□ "+A$:
   GOTO 114
80 IF T>7 THEN 100
82 IF T=6 THEN CLR:T=6
```



```
84 IFT>4THEN PRINT "◘"P$;
   "■FICHERO":INPUT F$:
   GOTO 100
86 F$=A$:IF T>2 THEN F$=
   "SUCESO"
88 PRINT CL$"ESCRIBE EL
   NUMERO DE■";F$:  PRINT
   "O CERO PARA SALIR"
90 INPUT U :U =INT(U):IFU=0
   THEN 50
92 IF U <1 OR (U>ZZ)THEN
   PRINT W$ (3):FOR DE =1 TO
   999:NEXT DE:GOTO 88
94 IFT>2THEN U=-U
96 GOSUB 450:CK=FA
98 IF(T=2 ORT=4)AND(O=U%(X)
   OR ZZ<U%(X)) THEN PRINT
   "NUMERO NO UTILIZADO":
   GOTO114
100 PRINT CL$:ON T GOSUB 120
    ,200,300,400,500,600,
    700,800,900,1000,2000,
    3000,960
101 IF KK$="S" AND(T>7 AND
    T<12)THEN OPEN 3,4:CMD3
105 IF KK$="S" AND(T>7 AND T
    <13)THEN PRINT#3:CLOSE 3
110 IF T<5 THEN 86
112 GOTO 50
114 FOR T=1 TO 1000:NEXT T:
    GOTO 50
120 IF 0<U%(X) AND ZZ>=U%(X)
    THEN GOSUB 942:GOSUB 932
    :GOTO 130
122 IF(AA=MA)THEN PRINT W$
    (2);F$:RETURN
124 AA=AA+1:A%(AA)=X:U%(X)=U
130 PRINT W$(4);F$:INPUT U$
    (X):XA=X
140 PRINT P$;"SUCESO INICIAL
    , SUCESO FINAL ":INPUT
    S,F:S=INT(S):F=INT(F)
142 IF S<1 OR S>ZZ OR F<1 OR
    F>ZZ THEN PRINT W$(3):
    GOTO 140
150 U=-S:GOSUB 450:IFU%(X)<0
    THEN 156
152 IF EE=ME THEN PRINT W$
    (2);"SUCESOS":GOTO 140
154 GOSUB350
156 S%(XA)=X
160 U=-F:GOSUB450:IFU%(X)<0
    THEN 166
162 IF EE=ME THEN PRINT W$
    (2);"SUCESOS":GOTO 140
164 GOSUB 350
166 F%(XA)=X
170 PRINTP$;"TIEMPO PROBABLE
    DE REALIZACION":INPUT
    T(XA)
172 IF T(XA)<0 THEN PRINT
    "NO LO PUEDES HACER TAN
    RAPIDO":GOTO 170
180 PRINTP$;"TIEMPO SEGURO
    AL 90%":PRINT"PARA SU
    REALIZACION":INPUT N(XA)
182 IF N(XA)<T(XA) THEN
    PRINT"ESTO DIFIERE DEL
    TIEMPO PROBABLE":
    GOTO 170
190 RETURN
200 FORB=1 TO AA:IFA%(B)=X
    THEN A=B
220 NEXTB:XX=A%(AA):U%(X)=ZZ
    +1:A%(A)=XX:AA=AA-1:
    RETURN
300 IF U%(X)<0 THEN PRINT
    "■SUCESOS":XP=U%(X):
    GOSUB 950:PRINT U$(X):
    GOTO 330
310 IF(EE=ME) THEN PRINT
    W$(2);F$:RETURN
312 GOSUB350
330 PRINTW$(4);F$:INPUT U$
    (X):S%(X)=0:RETURN
350 EE=EE+1:E%(EE)=X:S%(X)=
    -1:F%(X)=0:U%(X)=U
360 T(X)=0:N(X)=0:U$(X)="":
    RETURN
400 FOR F=1 TO EE:IF E%(F)=X
    THEN E=F
420 NEXT F:XX=E%(EE)::U%(X)=
    ZZ+1:E%(E)=XX:EE=EE-1:
    RETURN
450 Z=U-INT((U-1)/MH)*MH:
    Y=2:X=0
460 IF X=0 THEN IF 0=U%(Z)
    OR ZZ+1=U%(Z) THEN X=Z
470 IF U=U%(Z) THEN X=Z:
    RETURN
480 IF Y=1 OR 0=U%(Z) THEN
    RETURN
490 Z=Z+Y-MH*INT((Z+Y-1)/MH)
    :Y=Y+Y-MH*INT((Y+Y-1)
    /MH):GOTO460
500 OPEN 1,8,8,"0:"+F$+",
    SEQ,W":PRINT#1,MA;CM$;
    ME;CM$;MH;CM$;AA;CM$;EE;
    CM$;CK
```



```
510 IF CK THEN PRINT#1,SE;CM$;FE
520 FOR A=1 TO AA:X=A%(A):PRINT#1,X;
    CM$U%(X)CM$S%(X)CM$F%(X)CM$T(X)
    CM$N(X)CM$;
530 PRINT#1,G%(A)CM$;QT$;U$(X);QT$:
    NEXT A
540 FOR E=1 TO EE:X=E%(E):PRINT#1,X;CM$;
    U%(X);CM$;S%(X);CM$;F%(X);CM$T(X)
    CM$N(X)
550 PRINT#1,QT$;U$(X);QT$:NEXT E
560 FOR X=1 TO MH:IFU%(X)=ZZ+1 THEN
    PRINT#1,X
570 NEXT X:PRINT#1,0
580 CLOSE1:RETURN
600 OPEN 1,8,8,"0:"+F$+",SEQ,R":INPUT#1,
    MA,ME,MH,AA,EE,CK:GOSUB12
610 IF CK THEN INPUT#1,SE,FE
620 FOR A=1TOAA:INPUT#1,X,U%(X),S%(X),
    F%(X),T(X),N(X),G%(A),U$(X)
625 A%(A)=X:NEXT A
640 FOR E=1 TO EE:INPUT#1,X,U%(X),S%(X),
    F%(X),T(X),N(X),U$(X):E%(E)=X:
    NEXT E
650 INPUT#1,X:IFX>0 THEN U%(X)=ZZ+1:
    GOTO 650
660 CLOSE1:RETURN
700 OPEN15,8,15,"S0:"+F$:CLOSE15:RETURN
800 GOSUB 942
810 FOR A=1 TO AA:X=A%(A):GOSUB 932
820 Y=Y+1-(LEN(U$(X))>12):IF Y>20 AND
    (A>AA) THEN GOSUB 940:GOSUB 942
830 NEXT A :GOSUB 936:PRINT "  SUCESOS":
    PRINT:Y=3
840 FOR E=1 TO EE:X=E%(E):XP=U%(X):GOSUB
    950:PRINT U$(X)
850 Y=Y+1-(LEN(U$(X))>12):IF Y>20 AND
    E<EE THEN GOSUB 940:GOSUB 946
860 NEXT E:GOTO 936
900 INPUT"sREINICIO EL PROGRAMA (S/N)";
    AN$:IF AN$="N"THEN 50
910 IF AN$<>"S" THEN 900
920 RUN
932 XP=FNU(S%(X)):GOSUB 950:XP=FNU(F%
    (X)):GOSUB 950:XP=T(X):GOSUB 950:
    XP=N(X)
933 GOSUB 950:XP=ABS(U%(X)):GOSUB 950
935 PRINT:PRINT"TEXTO = ";U$(X):RETURN
936 IF KK$<>"S" THEN PRINT"  PULSA
    RETURN
    PARA CONTINUAR"
937 GET F$:IF F$=""THEN 937
938 IFF$<>CHR$(13)THEN 937
939 PRINT CLS$;:RETURN
941 RETURN
942 REM***CONTINUARA
```



EL ZOCO

Vendo los siguientes cartuchos para C-64: Dragonsden, Fútbol internacional, Sandra 64 mas su cassette con ejemplos y manual de instrucciones, o cambiaría por cartucho «Simon's Basic» u otros de utilidades como «L060», etc.

Juan José Martínez Mayordomo.
Carricola, 1, 2.ª
Tels. 331 28 37 y 377 44 26.
Valencia 46017.

• • • • •

Vendo Commodore 16 mas Datassette mas manual en inglés y en castellano mas juegos. Poseo múltiples revistas y libros.

Preguntar por Javier.
Tel. (91) 466 41 38.

• • • • •

Compro Commodore-64 de segunda mano en buen estado; ofertas a:

F. Javier García Coves
Calle 14, 6.
Tel. (965) 16 47 26.
Vistahermosa.
03015 Alicante.

• • • • •

Intercambio programas en disco o cinta.

Luis García Millán.
Avda. Valladolid, 25.
34004 Palencia.

• • • • •

Cambio programas de Commodore-64.

Pau Vaqués Mora.
Soledad, 104.
Igualada (Barcelona).

Vendo ordenador Commodore CBM-4032 y unidad doble de discos Commodore CBM-8050, todo ello en perfecto estado, con más de 100 programas, como Bases de Datos, Untilidades, Contabilidad y Juegos, así como todos los cables necesarios para su interconexión. Todo por 280.000 ptas. (400.000 de nuevo).

José Bermell Tarazona.
Plus Ultra, 23, 3.ª
Tel. (96) 334 03 40 (7 tarde en adelante).
46006 Valencia.

• • • • •

Desearía intercambiar en cinta o en disco todo tipo de programas para el C-64.

Xavier Rodríguez i Sagarra.
Pin i Soler, 6, 1.° B.
Tel. (977) 21 92 03.
43002 Tarragona.

• • • • •

Vendo Vic-20 en perfecto estado por tan solo 15.000 ptas. Además regalo varios libros y programas.

Ricardo Alba.
Sumatra, Q-2.
Tel. (952) 43 18 69.
El Atabal (Málaga 29010).

• • • • •

Club (C) LDA de Lérida. Intercambios a todo nivel con clubs y particulares sobre C-64.

Andreu Ibáñez Perales.
P.° de Ronda, 76, 3.°, 1.ª
Lérida 25006.

• • • • •

Intercambio programas, tengo unos 300.

Javier Hernández Sole.
El Roser, 34.
Tel. (977) 40 10 94.
Mora la Nova (Tarragona).

• • • • •

Vendo Vic-20 en perfecto estado, por 18.000 ptas.

Manuel Monge López.
Pintor López, 15
Tel (964) 21 78 65.
Castellón.

• • • • •

Cambio Spectrum Plus 64 K, mas interface Kemson programable, 40 juegos (Oue, On ohs Profanaicon, etc.), cables, dos manuales y las revistas, por Commodore CBM-64, manuales y cables. Pagaremos diferencia.

Tel. (91) 706 02 99.

• • • • •

Cambio programas C-64. Poseo más de 600 títulos (cassette).

Jordi Balaguer.
Príncipe de Viana, 84, 3.°
25008 Lérida.

• • • • •

Cambio programas Vic-20. Memoria original y 8 K.

José Luis Ripoll Bestard.
Legionario Aleixo, 8, 3.°
07011 Palma de Mallorca.
(Baleares.)

• • • • •

Desearía intercambiar programas en cinta para el Commodore-64.

Jesús Valdecasa Fombellida.
Lluch, 2, 4.°, 2.ª
08016 Barcelona.

• • • • •

Vendo Vic-20 casi nuevo; incluyo Datassette, curso Basic completo y extenso software (cintas y cartuchos).

Joan Al
Tel. (93) 798 66 29 (noches).

• • • • •

Me interesaría por unos libros donde se trata la tridimensión aplicada a la microinformática. Todo tipo de libros sobre el tema. Pagaría contra reembolso las fotocopias o cambiaría por otras. Si me pueden dar información sobre libros, también lo aceptaría.

Juan Ramón Elías Moreno.
Forjadores, 9.
Sevilla 41008.

• • • • •

Quiero formar un grupo de amigos del Commodore 64 en Majadahonda (Madrid).

Eduardo Torras Planas.
Beatriz de Silva, 2.
Tel. (91) 638 53 26.
Majadahonda (Madrid).

ACELERA TUS RUTINAS DE JUEGOS





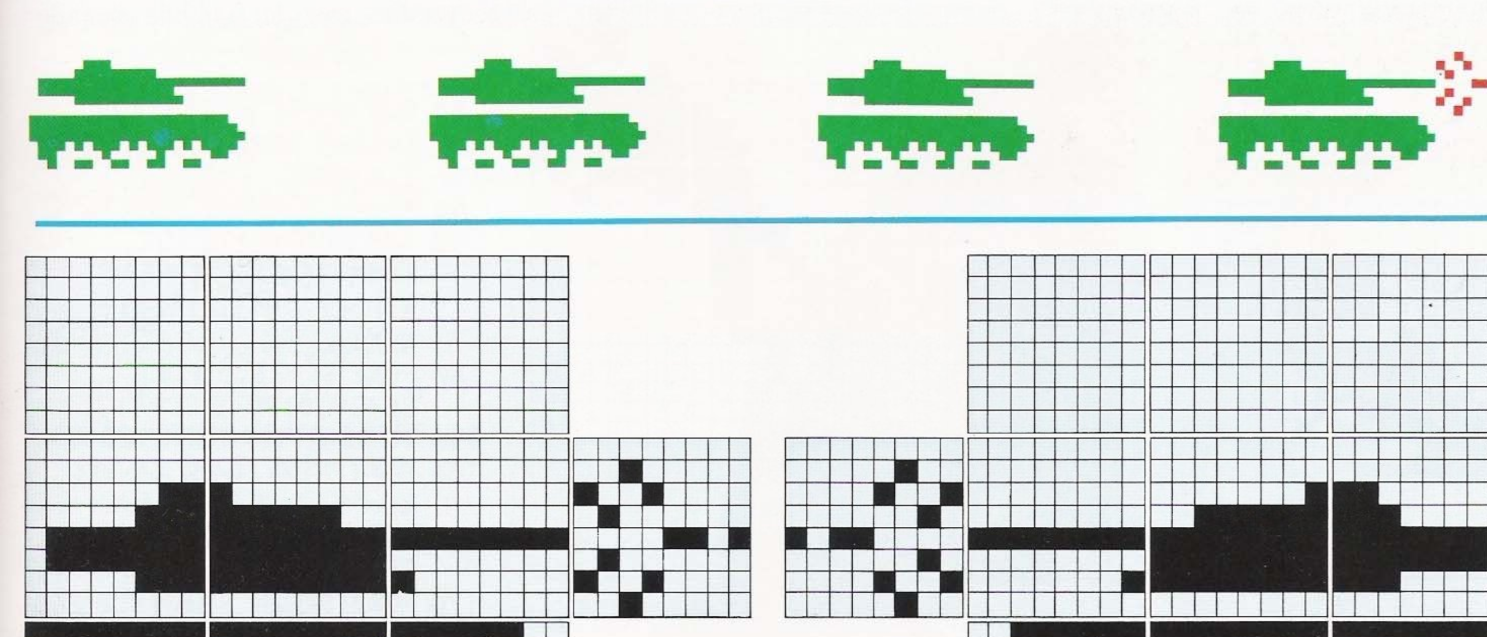


Figura 1.

El código máquina no es sólo para el programador experimentado. Tú puedes empezar con él, recurriendo a rutinas cortas que aceleren tus juegos en BASIC, y divertirte mientras aprendes.

La programación en código máquina dista mucho de ser fácil, al menos a simple vista. Para la mayoría de los usuarios de ordenadores personales, el código máquina no es más que un espantoso montón de números.

La mejor manera de comenzar es meterse a fondo con algunas rutinas escritas en código máquina. Así puedes familiarizarte, desde el principio, con sus ventajas y con lo que puede ofrecer a tus programas. Es por ello que hemos decidido llevarte a través de esos números misteriosos en este y en otros próximos artículos, de tal manera que puedas ser capaz de desarrollar programas en código máquina por ti mismo.

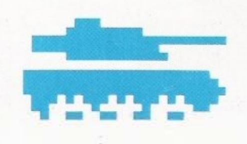
Los gráficos que van en este artículo utilizan programas en BASIC para depositar las rutinas en código máquina en la memoria de tu ordenador. De esta manera puedes producir movimientos mucho más rápidos y con más apariencia de vida de lo que podrías conseguir utilizando solamente el BASIC.

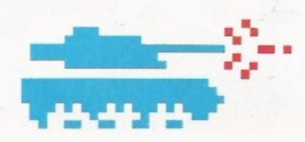
La mejor manera de crear gráficos similares a los de las figuras 1 y 2 es utilizar una de las características gráficas especiales del **Commodore 64:** los *Sprites,* el rey de los GDU.

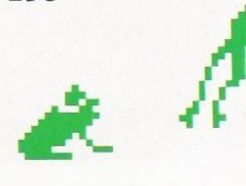
CREA EL TANQUE

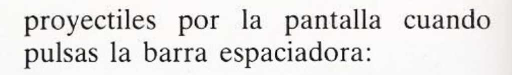
Este programa de *Sprites* cubre tres actividades: una rutina de inicialización, una secuencia de control, que regula el movimiento del **Sprite** por la pantalla, y los datos utilizados para definir la forma del *Sprite*-tanque:

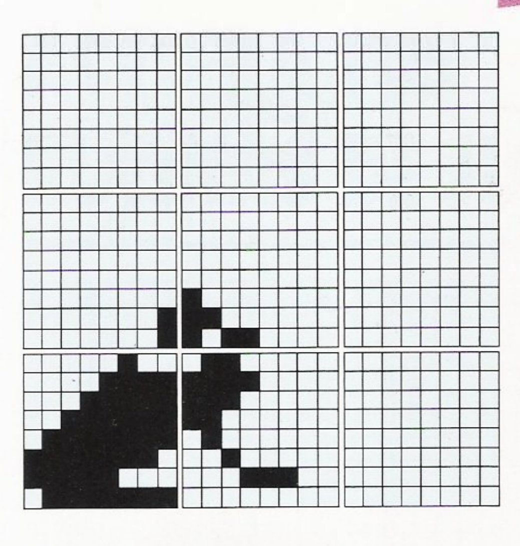
```
10 FOR M=252 TO 253:FOR
   I=64*M TO 64*M+63:POKE
   I,0:NEXT
20 FOR I=64*M+21 TO
   64*M+56:READ A:POKE
   I,A:NEXT:NEXT
30 SC=53248:X3=24:X=24:
   Y3=157:Y=157
40 POKE 2043,252:POKE
   2042,254:POKE SC
   +23,0
50 POKE SC+29,0:POKE
   SC+42,12:POKE SC
   +27,0
90 J=0:R=0:POKE
   650,128:PRINT"♥":POKE
   SC+21,8
100 GET A$:IF A$=""THEN 180
110 IF A$="Z" AND X3>24
    THEN X3=X3-12
120 IF A$="X" AND X3<310
    THEN X3=X3+12
130 IF A$="P" AND Y3>50
    THEN Y3=Y3-12
140 IF A$="L" AND Y3<220
    THEN Y3=Y3+12
180 IF X<X3 THEN X=X+3:POKE
    2043,252
190 IF X>X3 THEN X=X-3:POKE
    2043,252
200 IF Y<Y3 THEN Y=Y+3
210 IF Y>Y3 THEN Y=Y-3
220 XA=INT(X/256):XB=
    X-XA*256
```











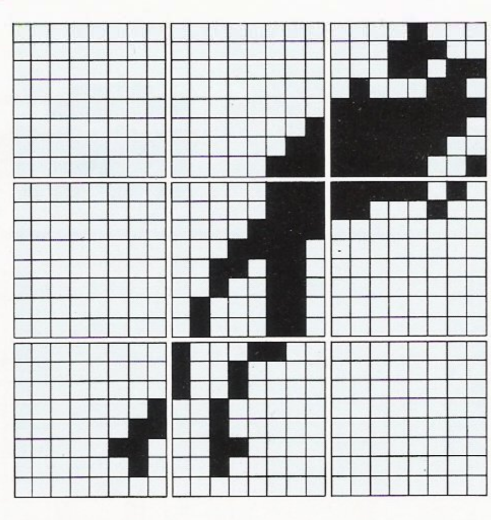


Figura 2. Esquema de la composición de la rana.

```
230 POKE SC+6,XB:POKE
    SC+7,Y
240 POKE SC+16,4*(1 AND
    JA)+8*(1 AND XA)
250 GOTO 100
1000 DATA 0,1,192,0,63,224,
     255,255,254,0,255,
     254,1,255,224,0,0,0
1010 DATA 63,255,255,127,
     255,255,255,255,255,
     122,235,174,48,65,6
1020 DATA 6,102,100
1030 DATA 3,128,0,7,252,0,
     127,255,255,127,255,0,
     7,255,128,0,0,0
1040 DATA 255,255,252,255,
     255,254,255,255,255,
     117,215,94,96,
     130,12
1050 DATA 38,
     102,96
```

Si ahora ejecutas el programa (RUN), verás un tanque desplazarse por la pantalla, bajo las órdenes de las teclas Z y X, que lo mueven de izquierda a derecha, y P y L, que hacen lo propio de arriba a abajo. Para detener el programa se recurre a la tecla RUN/STOP. Advierte cómo el tanque cambia de forma cuando se altera la dirección del movimiento.

AÑADELE ACCION

Hasta ahora sólo has utilizado uno de los *Sprites* disponibles, pero puede crearse otro para dotarle de fuego al tanque. Este programa dispara proyectiles por la pantalla cuando pulsas la barra espaciadora:

```
30 FOR M=254 TO 255:FOR
   I=64*M TO 64*M+62:POKE
   I,0:NEXT
40 M2=0:IF M=255 THEN M2=2
50 FOR I=64*M+18 TO
   64*M+36 STEP 3:READ
   A:POKE I+M2,A:NEXT:
   NEXT
80 POKE SC+23,0:POKE
   SC+29,0:POKE SC+41,1
150 IF A$<>"□" THEN 180
160 RT=(PEEK(2043)=253):
    X2=-360*RT:Y2=Y:J=X-24
    *RT-11:POKE 2042,
    254-RT
170 POKE SC+21,12:GOSUB 260
260 FOR T=J TO X2-4
    STEP(RT*2+1)*-20
270 JA=INT(T/256):JB=
    T-JA*256
280 POKE SC+16,4*(1 AND
    JA)+8*(1 AND XA)
290 POKE SC+4,JB:POKE
    SC+5,Y
300 FOR P=1 TO
    30:NEXT:NEXT:RETURN
1500 DATA 4,9,2,176,2,9,4
1510 DATA 32,144,64,13,64,
     144,32
```

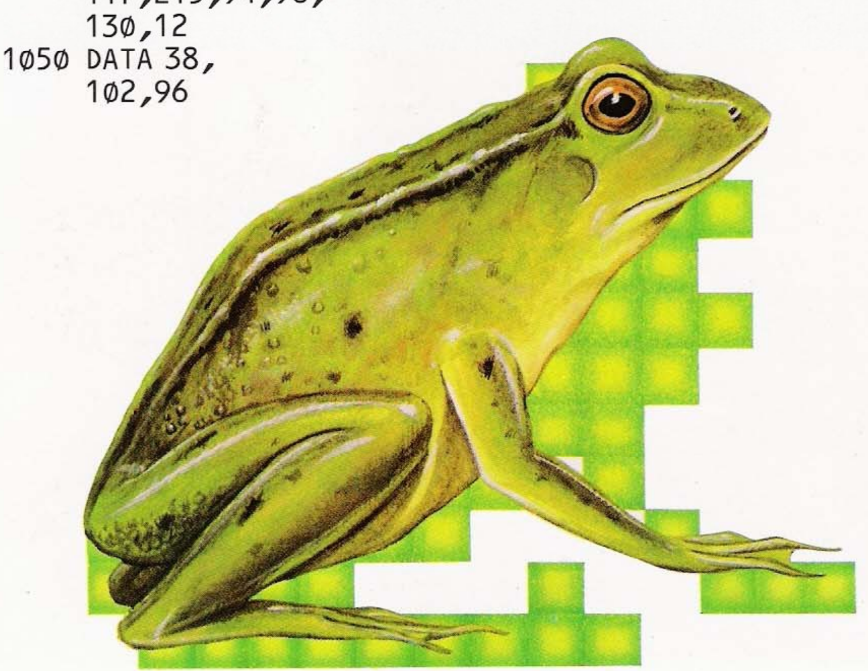
Para guardar (SAVE) el programa de cara a futuras utilizaciones, teclea primero POKE 53269,0 (que limpia de *Sprites* la pantalla) y después emplea el SAVE normalmente utilizado en BASIC.

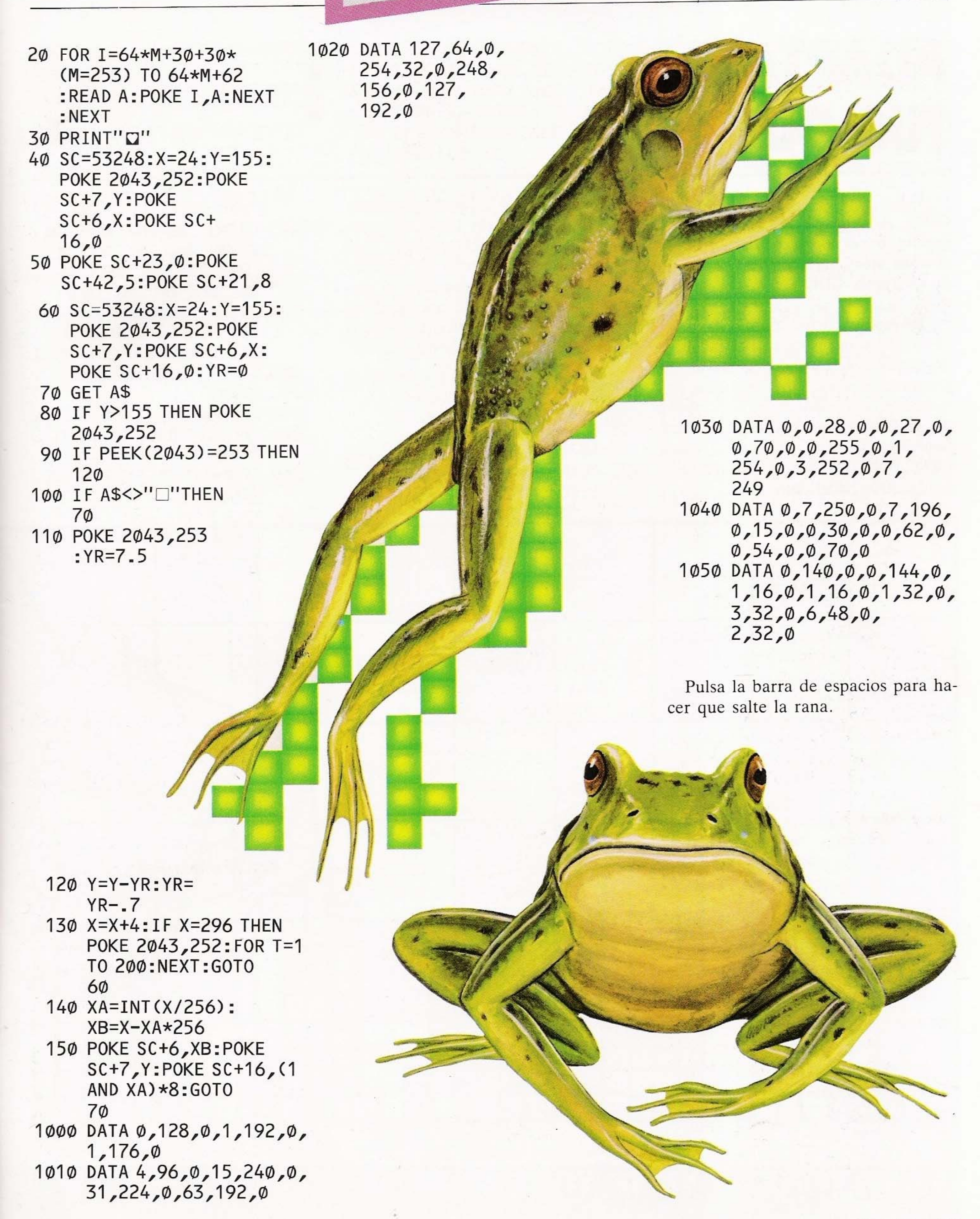
CREACION DE LA RANA

Intentémoslo ahora con la rana. El gráfico de la misma aparece en la figura 2, pero debe ser ligeramente alterado para poder ser acomodado en un solo *Sprite*. Esto se hace quitando tres de las cuatro filas. El único efecto es que las piernas se acortan ligeramente.

Desconecta el ordenador, o simplemente teclea NEW RETURN antes de introducir este programa:

```
10 FOR M=252 TO 253:FOR
   I=64*M TO 64*M+29:POKE
   I,0:NEXT
```





```
20 FOR I=64*M+3Ø+3Ø*
   (M=253) TO 64*M+62
   :READ A:POKE I,A:NEXT
   :NEXT
3Ø PRINT"♥"
4Ø SC=53248:X=24:Y=155:
   POKE 2Ø43,252:POKE
   SC+7,Y:POKE
   SC+6,X:POKE SC+
   16,Ø
5Ø POKE SC+23,Ø:POKE
   SC+42,5:POKE SC+21,8
6Ø SC=53248:X=24:Y=155:
   POKE 2Ø43,252:POKE
   SC+7,Y:POKE SC+6,X:
   POKE SC+16,Ø:YR=Ø
7Ø GET A$
8Ø IF Y>155 THEN POKE
   2Ø43,252
9Ø IF PEEK(2Ø43)=253 THEN
   12Ø
1ØØ IF A$<>"□"THEN
   7Ø
11Ø POKE 2Ø43,253
   :YR=7.5
12Ø Y=Y-YR:YR=
   YR-.7
13Ø X=X+4:IF X=296 THEN
   POKE 2Ø43,252:FOR T=1
   TO 2ØØ:NEXT:GOTO
   6Ø
14Ø XA=INT(X/256):
   XB=X-XA*256
15Ø POKE SC+6,XB:POKE
   SC+7,Y:POKE SC+16,(1
   AND XA)*8:GOTO
   7Ø
1ØØØ DATA Ø,128,Ø,1,192,Ø,
   1,176,Ø
1Ø1Ø DATA 4,96,Ø,15,24Ø,Ø,
   31,224,Ø,63,192,Ø
1Ø2Ø DATA 127,64,Ø,
   254,32,Ø,248,
   156,Ø,127,
   192,Ø
1Ø3Ø DATA Ø,Ø,28,Ø,Ø,27,Ø,
   Ø,7Ø,Ø,Ø,255,Ø,1,
   254,Ø,3,252,Ø,7,
   249
1Ø4Ø DATA Ø,7,25Ø,Ø,7,196,
   Ø,15,Ø,Ø,3Ø,Ø,Ø,62,Ø,
   Ø,54,Ø,Ø,7Ø,Ø
1Ø5Ø DATA Ø,14Ø,Ø,Ø,144,Ø,
   1,16,Ø,1,16,Ø,1,32,Ø,
   3,32,Ø,6,48,Ø,
   2,32,Ø
```

Pulsa la barra de espacios para hacer que salte la rana.

ADAPTA TUS GDU (y II)

- NUEVOS CONTROLES
- GDU INVERSOS
- GDU GIRATORIOS
- USO DE LOS GDU EN TUS PROGRAMAS

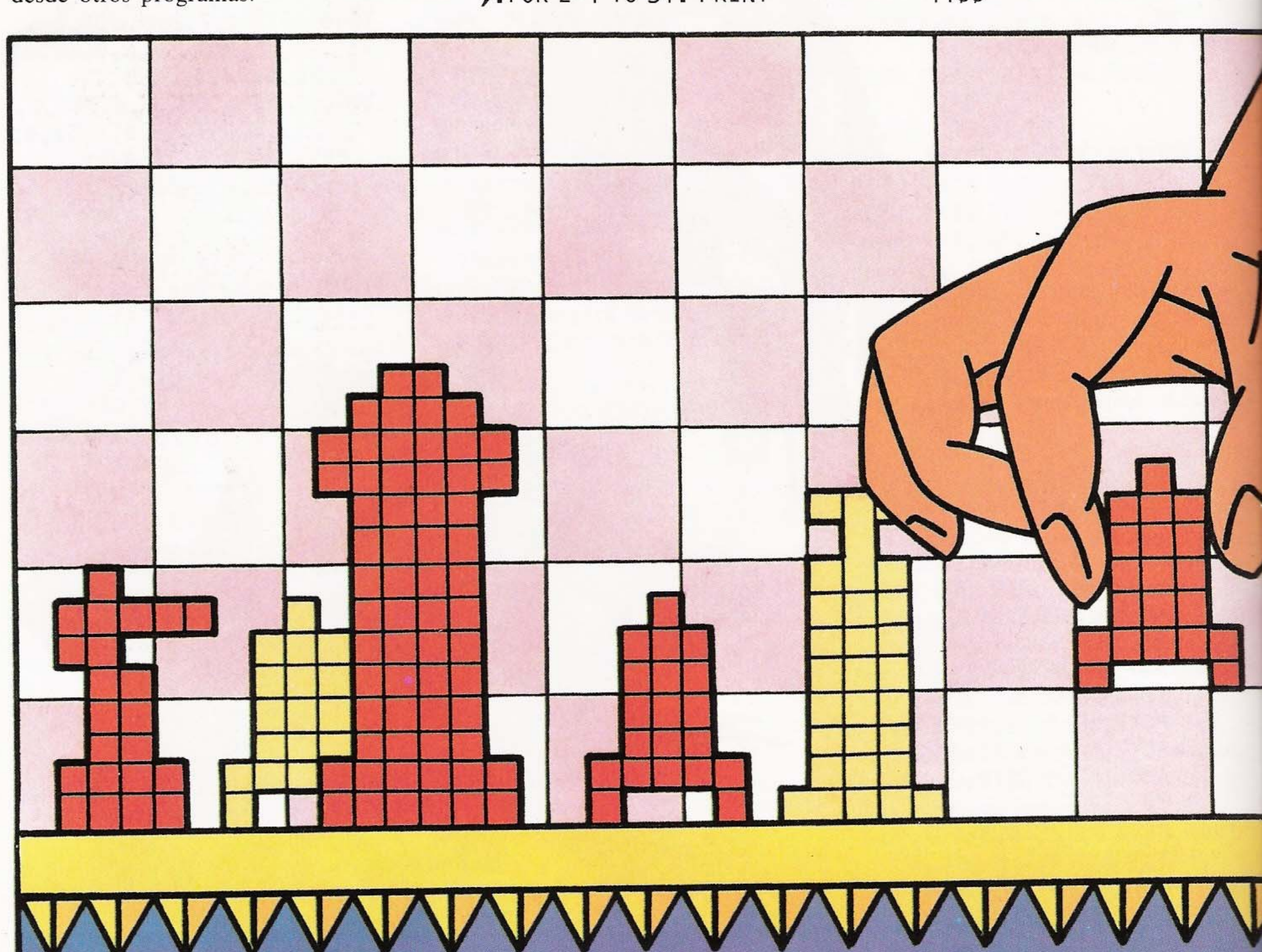
¿Falta algo en tu generador de GDU? A continuación te ofrecemos nuevas líneas de programa que te abrirán otros horizontes en el diseño de excitantes GDU.

En este artículo presentamos el resto de los programas del generador de caracteres. Las nuevas líneas de programa añaden varias mejoras, para que te resulte aún más fácil diseñar tus propios caracteres gráficos. Hay también una guía para utilizar los GDU definidos por ti, llamándolos desde otros programas.

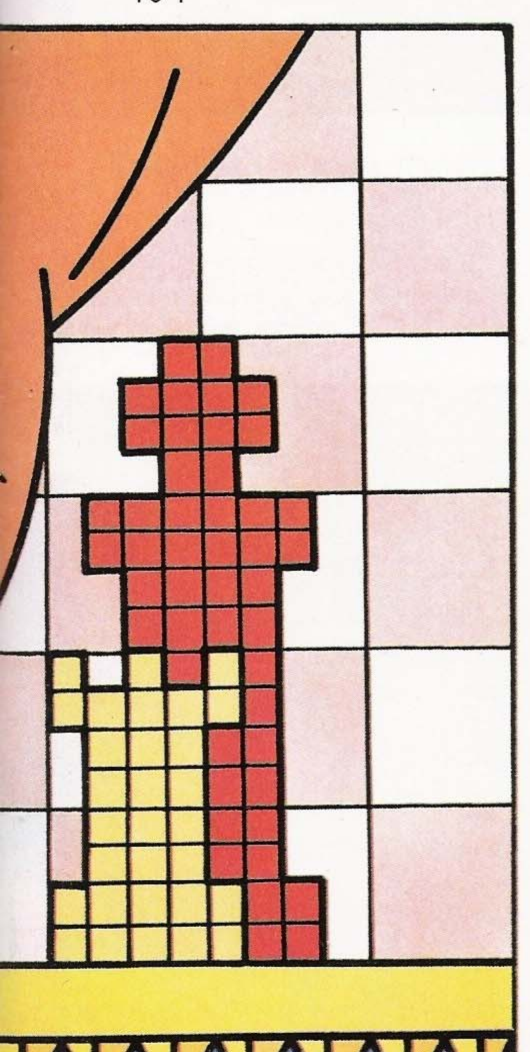
Las siguientes líneas de programa añaden unas cuantas funciones nuevas al generador de caracteres que publicamos en el número de septiembre. Si guardaste el programa en cinta, puedes cargarlo de nuevo ahora con LOAD. Si no lo hiciste, tecléalo incorporándole las siguientes líneas.

Teclea para Commodore-64

```
33 FOR Z=Ø TO 7:POKE
   12288+Z,255:NEXT Z
123 PRINT "▣▣▣▣▣▣▣▣▣▣▣◸"
   ;:FOR Z=1 TO 31: PRINT
   "⊟";:NEXT Z: PRINT "◹"
125 PRINT "▣▣▣▣▣◫▬▣@□▣
   @□▣@□◣@□▩@□▣@□▣@
   □π@□♠@□◢@□⊠@□●@□
   ♣@□▮@□◆@□▚@◆▣◫"
128 PRINT "▣▣▣▣◹";:FOR
   Z=1 TO 31:PRINT
   "⊟";:NEXT Z:PRINT "◸"
165 IF P=55 AND Y<7 THEN
   Y=Y+1
21Ø IF P=4 THEN 1ØØØ
22Ø IF P=5 THEN FL=1:GOTO
   11ØØ
23Ø IF P=6 THEN FL=2:GOTO
   11ØØ
```



```
240 IF P=3 THEN FL=3:GOTO
    1100
245 IF P=51 THEN FL=4:GOTO
    1120
260 IF P=49 THEN A=0:GOTO
    1230
270 IF P=54 THEN 1300
1000 FOR Z=0 TO 7:FOR ZZ=0
     TO 7
1010 C1=1396+ZZ+Z*40:
     C2=PEEK(C1)
1020 IF C2=207 THEN POKE
     C1,230
1030 IF C2=230 THEN POKE
     C1,207
1040 NEXT ZZ,Z:GOTO 130
1100 FOR Z=0 TO 7:FOR ZZ=0
     TO 7
1110 C1=1396+ZZ+Z*40:
     C(Z,ZZ)=PEEK(C1): NEXT
     ZZ,Z
1120 FOR Z=0 TO 7:FOR ZZ=0
     TO 7
```



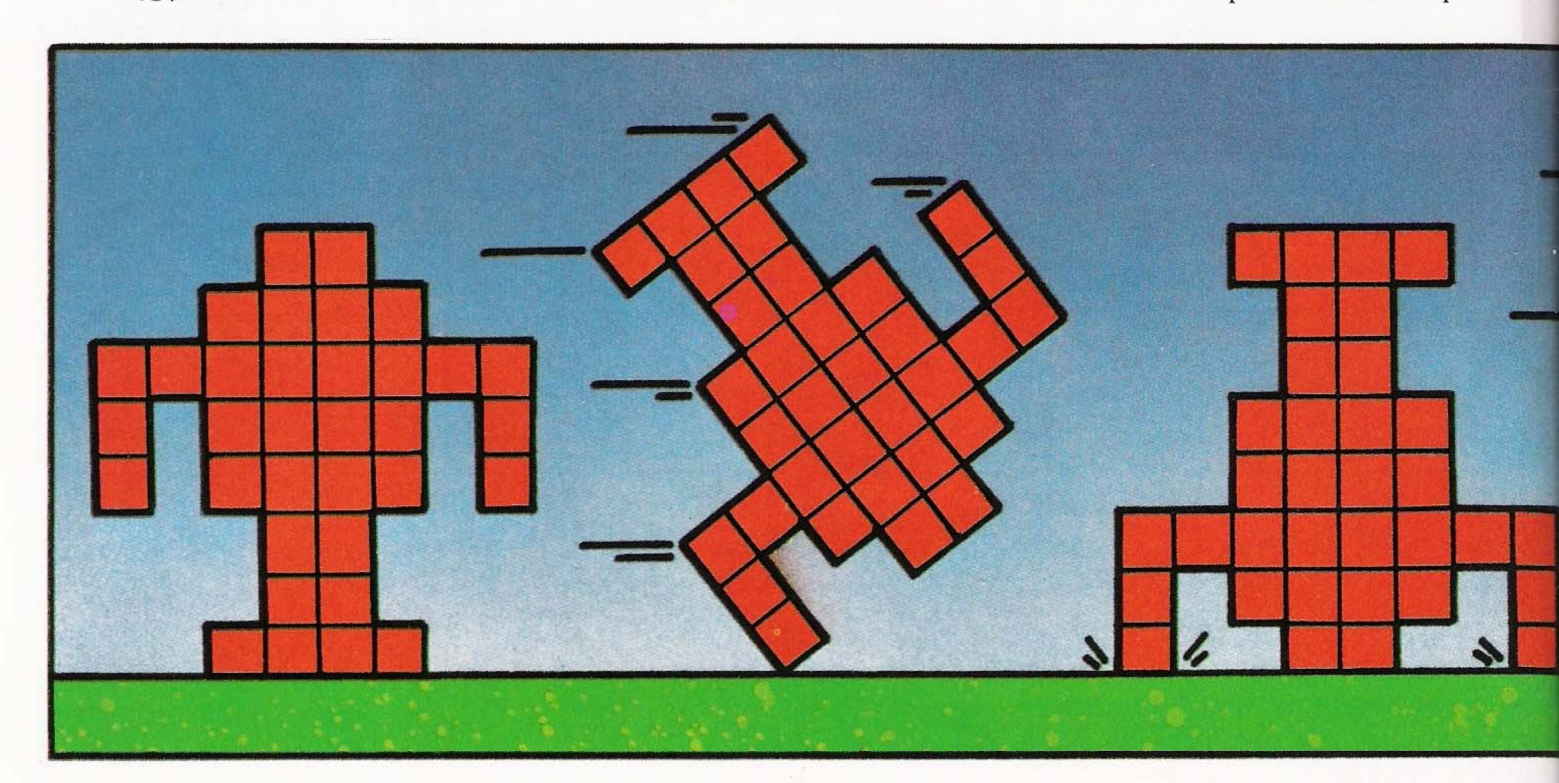
```
1130 IF FL=1 THEN POKE
     1399+ZZ+Z*40,C(ZZ,7-Z)
1132 IF FL=2 THEN POKE
     1399+ZZ+Z*40,C(7-ZZ,Z)
1134 IF FL=3 THEN POKE
     1399+(7-ZZ)
     +Z*40,C(Z,ZZ)
1136 IF FL=4 THEN POKE
     1399+ZZ+Z*40,207
1140 NEXT ZZ,Z
1145 IF FL=4 THEN A=0: GOTO
     1230
1150 GOTO 130
1230 PRINT "■■■■■■■■■■"
1270 PRINT "■";SPC(24);Z+1;
     "⊟□□□□▮▮▮▮▮▮";T
1300 PRINT "■■■■■";
     TAB(9); "(D)ATOS O
     (C)ARACTERES?": FOR Z=1
     TO 25: NEXT Z
1301 PRINT "■■■■■";
     TAB(9);"□□□□□□□□□□
     □□□□□□□□□□□□□□"
1302 GET A$:IF A$<>"D" AND
     A$<>"C" THEN 1300
1303 OPEN 4,4:CMD4:IF A$="D"
     THEN 1700
1304 FOR Z=1 TO 26:PRINT
     SPC(6); CHR$(64+Z): FOR
     ZZ=0 TO 7: PRINT ZZ+1;
1305 A=PEEK(12288+Z*8+ZZ)
1310 AA=A:FOR Z1=0 TO 7:IF
     A-Z(Z1)=>0 THEN A=A-
     Z(Z1): PRINT "■□■";:
     GOTO 1320
1315 PRINT ".";
1320 NEXT Z1:PRINT AA:NEXT
     ZZ:PRINT:NEXT Z
1330 PRINT#4:CLOSE4:GOTO
     130
1620 NEXT Z1,ZZ: GOTO 1230
1700 FOR Z=1 TO 26:PRINT
     CHR$ (64+Z);"=";:FOR
     ZZ=0 TO 7
1710 PRINT PEEK
     (12288+Z*8+ZZ);:NEXT
     ZZ:PRINT:NEXT Z
1720 PRINT#3:CLOSE4:GOTO 130
```

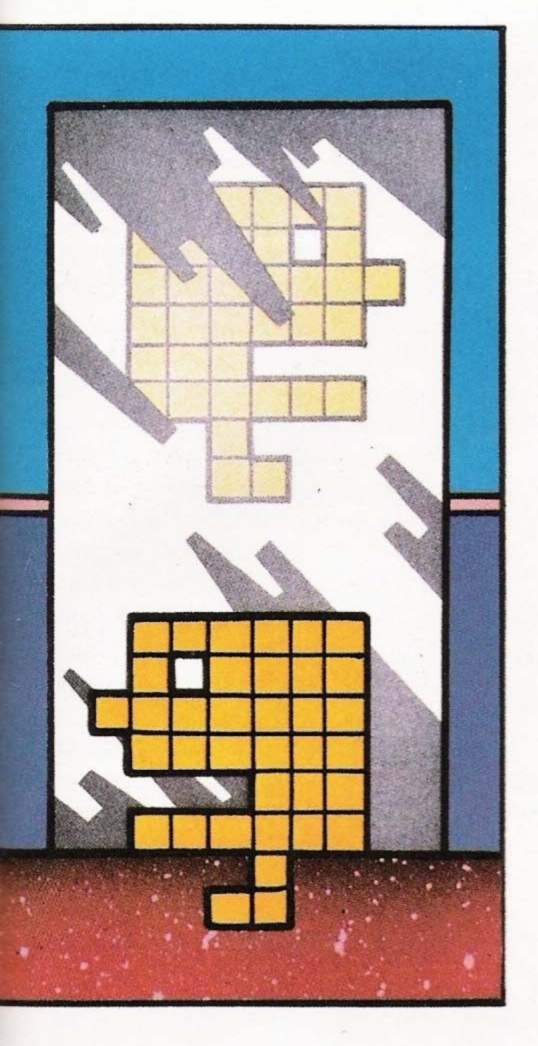
Teclea para Vic-20

```
125 PRINT "■■■■■■■■■■
    ■■@□■@□■@□◣@□▓@□
    ■@□■@□■@□■"
210 IF P=39 THEN 1000
220 IF P=47 THEN FL=1:GOTO
    1100
230 IF P=55 THEN FL=2:GOTO
    1100
240 IF P=63 THEN FL=3:GOTO
    1100
245 IF P=62 THEN FL=4:GOTO
    1230
260 IF P=14 THEN A=0:GOTO
    1230
270 IF P=54 THEN 1300
1000 FOR Z=0 TO 7:FOR ZZ=0
     TO 7
1010 C1=7880+ZZ+Z*22:
     C2=PEEK(C1)
1020 IF C2=207 THEN POKE
     C1,230
1030 IF C2=230 THEN POKE
     C1,207
1040 NEXT ZZ,Z:GOTO 130
1100 FOR Z=0 TO 7:FOR ZZ=0
     TO 7
1110 C1=7880+ZZ+Z*22:C(Z,ZZ)
     =PEEK(C1):NEXT ZZ,Z
1120 FOR Z=0 TO 7:FOR ZZ=0
     TO 7
1130 IF FL=1 THEN POKE
     7880+ZZ+Z*22,C(ZZ,7-Z)
1132 IF FL=2 THEN POKE
     7880+ZZ+Z*22,
     C(7-ZZ,Z)
1134 IF FL=3 THEN POKE
     7880+(7-ZZ)
     +Z*22,C(Z,ZZ)
1136 IF FL=4 THEN POKE
     7880+ZZ+Z*22,207
1140 NEXT ZZ,Z
1145 IF FL=4 THEN A=0: GOTO
     1230
1150 GOTO 130
1230 PRINT "■■■■■■■■■■"
1270 PRINT "■";SPC(12);
     Z+1;"⊟□□□□▮▮▮▮▮▮";T
1300 PRINT "■■■■■(D)ATOS O
     (C)ARACTERES?"
1301 PRINT "■■■■■□□□□
     □□□□□□□□□□□□□□□
     □□□"
1302 GET A$:IF A$<>"D" AND
     A$<>"C" THEN 1300
1303 OPEN 4,4:CMD4:IF A$="D"
     THEN 1700
1304 FOR Z=1 TO 22:PRINT
     SPC(6); CHR$(64+Z):FOR
     ZZ=0 TO 7: PRINT ZZ+1;
1305 A=PEEK(7168+Z*8+ZZ)
1310 AA=A:FOR Z1=0 TO 7: IF
     A-Z(Z1)=>0 THEN A=A-
     Z(Z1): PRINT "■□■";:
```

```
     GOTO 132Ø
1315 PRINT ".";
132Ø NEXT Z1:PRINT AA:NEXT
     ZZ:PRINT:NEXT Z
133Ø PRINT#4:CLOSE4:GOTO
     13Ø
17ØØ FOR Z=1 TO 22:PRINT
     CHR$(64+Z);"=";:FOR
     ZZ=Ø TO 7
171Ø PRINT PEEK
     (7168+Z*8+ZZ);:NEXT
     ZZ:PRINT:NEXT Z
172Ø PRINT#4:CLOSE4:GOTO
     13Ø
```

Si has tecleado las nuevas líneas, ya está completo tu generador de caracteres. Ahora puedes moverte por la





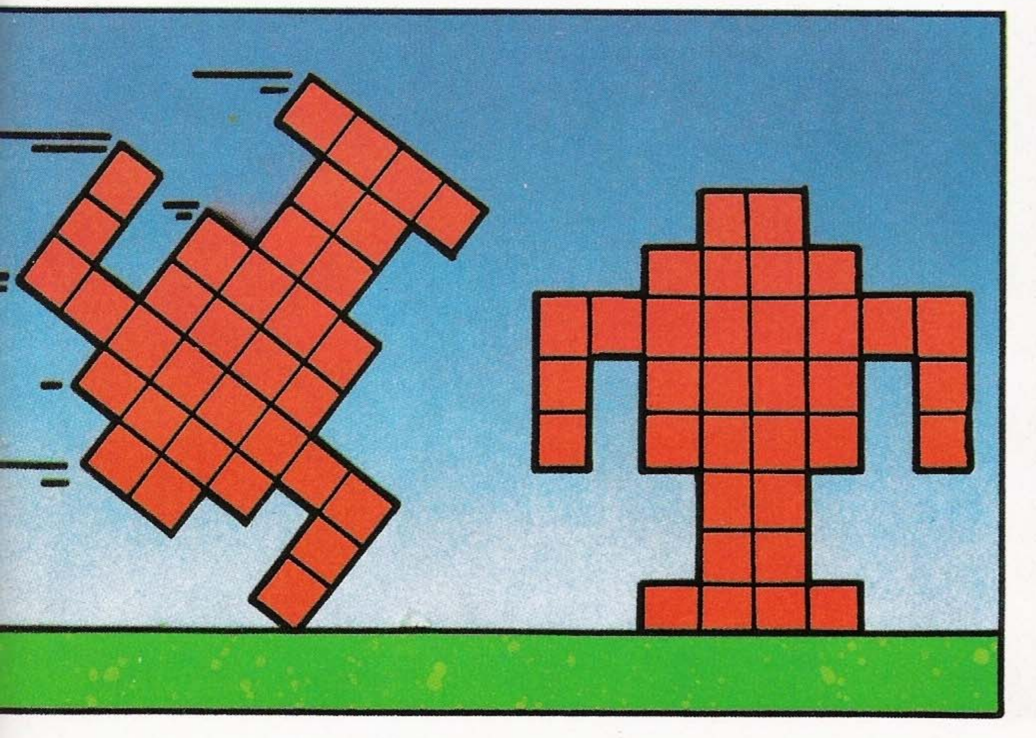
retícula, poner y quitar elementos de imagen *(pixels)*, así como girar la rejilla, obtener su imagen espectacular, invertirla y borrarla. Para varios de estos nuevos controles se utilizan las teclas de las funciones.

Para girar la retícula hacia la izquierda, pulsa F3 y para la derecha, F5. La posibilidad de hacer girar un GDU resulta útil cuando se quieren tener caracteres iguales o muy parecidos, orientados según direcciones diferentes, tal como ocurre en una nave espacial cuando la haces girar para disparar sobre cualquier punto de la pantalla.

La rutina de reflexión especular, que se obtiene pulsando F7, presenta la imagen de la retícula reflejada en un espejo de izquierda a derecha, es decir, con simetría respecto a un eje vertical que pase por el centro de la retícula. Puedes utilizarla para producir figuras simétricas que incluyan dos GDU.

Con la tecla F1 se invierte el dibujo, lo que quiere decir que si un punto de imagen, o *pixel,* estaba encendido, ahora está apagado y viceversa. Esto puede dar resultados sorprendentes.

Si después de construir un GDU en la retícula, te parece que no es lo que tú querías, puedes borrarlo del todo, borrando la retícula y dejándola lista para empezar otra vez. Esto se hace con la tecla CLR/HOME. Como esta tecla está lejos de las teclas de movimiento, es poco probable que la aprietes por accidente.



USO DE TUS GDU

La primera parte del programa contenía una rutina de almacenamiento en cinta, con la que puedes almacenar tus GUDU para usarlos en tus propios programas. También puedes usar los valores de los datos en cada diseño. Puedes copiar a mano los datos de la pantalla, pulsando la tecla de asterisco (*) sobre la retícula de cada GDU, pero esto te puede llevar mucho tiempo. Si tienes impresora, también puedes imprimir los datos con ella. Para hacerlo, pulsa la tecla de flecha hacia arriba.

Si quieres almacenar los caracteres en cinta y usarlos en tus propios programas sin teclear los DATAs, tienes que cargarlos de nuevo. Desgraciadamente esto no es tan fácil como podría pensarse, y tienes que usar la rutina de carga de las líneas 1410 a 1420.

No te olvides que en la línea 1410 sólo has de utilizar la mitad que se aplica a tu caso: la primera, si tienes cinta, la segunda, si tienes *diskette.* Reemplaza en esta línea la variable OU con 0, y la cadena variable N$ con el nombre del fichero almacenado (el nombre que habrás tecleado al guardar con SAVE el conjunto de caracteres del generador de GDU.

Organizando la memoria de tu ordenador, puedes tener a la vez varios conjuntos de caracteres. Puedes cambiar la zona de memoria en que se cargaron los datos, para que puedas almacenar varios bloques de caracteres desde el generador de GDU y, a continuación, cargarlos en memoria.

Para hacer esto, has de cambiar los números del bucle FOR...NEXT que hay en la línea 1413. El primero de dichos números indica dónde empiezan dichos caracteres en memoria, y el segundo número indica dónde terminan. También puedes modificar los punteros de caracteres, cambiando los POKEs de la línea 10.

Nota: En algunos **Vic-20** si se utilizan sin cartucho de ampliación al teclear este programa, pueden aparecer mensajes de error por falta de espacio en la memoria.

¿COMMODORE 64 O SPECTRUM?

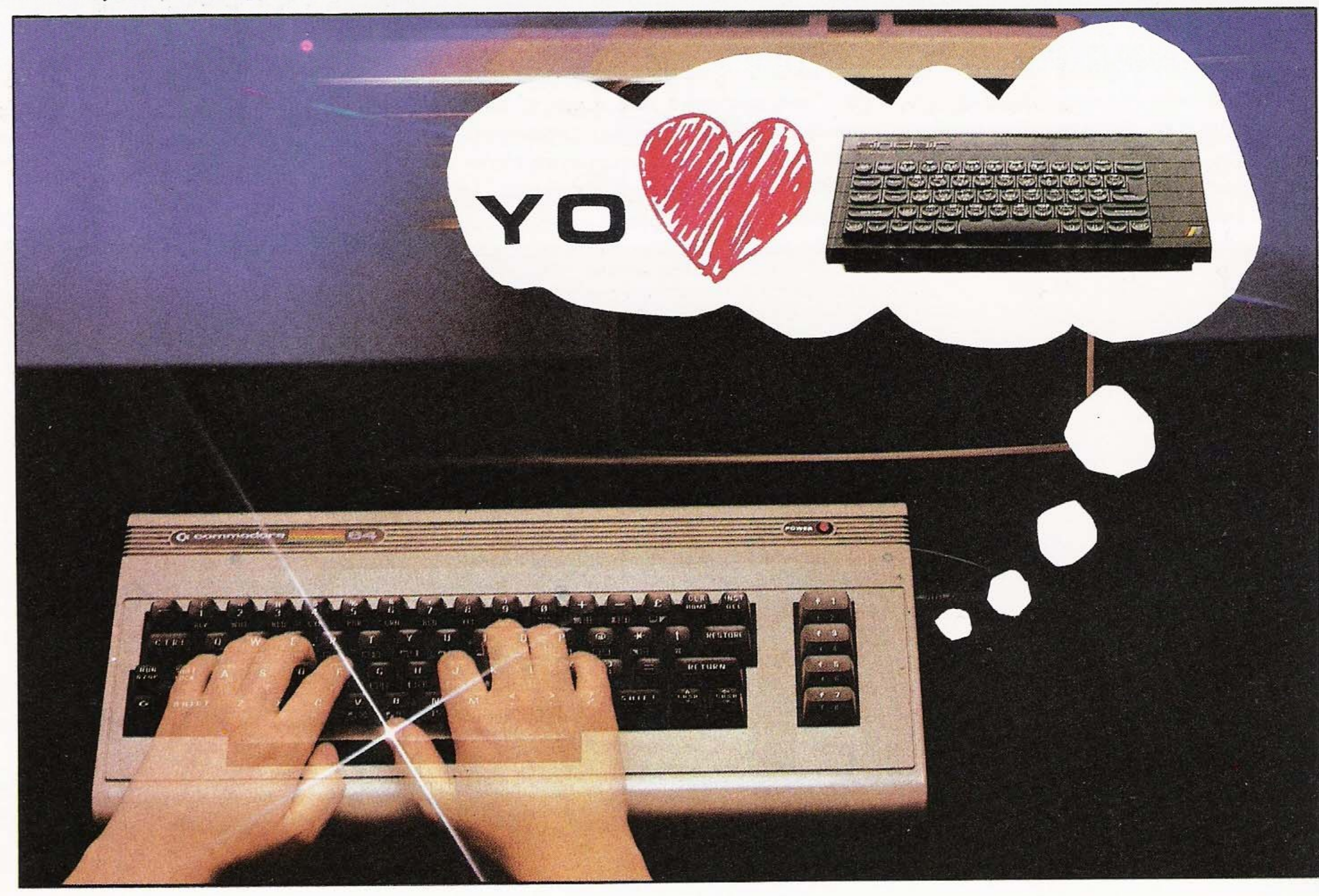
No cabe la menor duda de que el Commodore 64 **es un ordenador tremendamente polifacético. Incorpora prestaciones magníficas en los campos gráfico y sonoro; pero, además, es capaz de trabajar como un** Spectrum **de** Sinclair. **Al menos con programas en BASIC.**

Esta última aseveración se hace realidad por obra y gracia de un programa en *cassette,* que no es otra cosa que un simulador del funcionamiento del otro popular microordenador. Una vez cargado en el **64,** es posible teclear un programa destinado al **Spectrum,** sin mayores complicaciones de las que pueden aparecer mientras se trabaja con la máquina de **Sinclair.**

Una vez que el programa está en marcha, la sensación de haber cambiado de ordenador es real. La pantalla permanece toda ella en blanco y el cursor con la letra K en inverso, parpadea en la parte inferior, esperando a que escribamos un comando directo o un número de línea.

Vamos a ver qué ocurre si obligamos al ordenador a realizar un bucle FOR...NEXT. Tecleamos FOR...; , pero, ¡cáspita!, lo que aparece es FORor. Lo que pasa es sencillo: acabamos de entrar en el mundo de las palabras clave «tokenizadas». En algún momento hemos dicho en **INPUT** que el ordenador, a través de su editor e intérprete del BASIC, traduce determinadas palabras clave (tales como REM, PRINT, FOR, etc.) a un equivalente que ocupa un solo byte en la memoria —en lugar de uno por letra, como sería lo más lógico—. Este proceso de ahorro de memoria se llama «tokenización». En el **Spectrum** estas palabras aparecen impresas en las teclas y es posible escribirlas de una vez, mostrándose en la pantalla como si las hubiéramos escrito letra a letra.

Las teclas del ordenador de **Sinclair** son, por tanto, polivalentes. Es decir, la presión de una tecla nos permite acceder a varias posibilidades, en función de la letra que aparezca en inverso dentro del cursor. Así, la K indica que espera un comando, la G un comando gráfico, la L una letra, etc. La presión de una determinada tecla se relaciona con la leyenda de ese cursor. En el **64** ocurre ahora lo mismo, sólo que deberemos fijarnos en el diagrama del teclado de un **Spectrum** para no desorientarnos, puesto que la correspondencia entre teclas es casi idéntica. Lo de casi quiere decir que



ESTE ES EL SIMBOLO DE COMMODORE, COMPAÑIA AMERICANA, LIDER MUNDIAL EN NUMERO DE ORDENADORES INSTALADOS.

Este PC de Commodore ha hecho dudar a más de uno.

La oferta del mercado de los ordenadores PC dejaba hasta hoy muy sencilla la elección. Sin embargo, Commodore, líder reconocido en varios sectores de la informática, ha ofrecido una respuesta alternativa que atiende plenamente las exigencias empresariales y de profesionales liberales: su nuevo ordenador PC.

El nuevo Commodore PC dispone de una versatilidad acorde con una tecnología depurada en constante evolución y compatible con el software standard que más le suena.

commodore PC

Estos avances, y un precio realmente interesante, han planteado serias dudas entre los profesionales más cualificados a la hora de elegir un buen PC.

Sin duda Commodore, con el mayor número de ordenadores vendidos en el mundo se afianza en el campo empresarial con mucha fuerza.

Si está interesado en conocer más de cerca el nuevo PC de Commodore, pregunte en cualquier concesionario Commodore, le sacará de dudas.

PRINCIPALES CARACTERISTICAS

– 256 K de RAM de 9 bits – Zócalos para 512 K más – 2 unidades de diskete de 360 K – Disco 10 Mb opcional – Interfases serie y pararelo, incluidos – 5 slots compatibles – El mejor precio en esta categoría.



Microelectrónica y Control *c/ Valencia, 49-53 08015 Barcelona - c/ Ardemans, 8 28028 Madrid*
Unico representante de Commodore en España.

algunas de las teclas más genéricas del **64** siguen conservando su función, por ejemplo: los movimientos de cursor; RETURN, que pasa a ser el equivalente de ENTER, etc.

La edición de una línea de programa previamente escrita se consigue mediante CLR/HOME que escribe la línea en la base de la pantalla y nos permite desplazarnos por ella con las teclas de movimiento del cursor.

La otra edición, la normal del **Spectrum,** que indica los errores antes de autorizar la inserción de la línea en un programa, también ha sido incorporada en el emulador. Si presionamos RETURN, perdón, ENTER, aparece una interrogación parpadeante tras lo erróneamente escrito. Este tipo de detección de errores sería de gran ayuda en otros ordenadores, **Commodore** incluidos, porque permite ahorrar gran cantidad de «sintax errors» mientras ejecutamos un programa.

La principal limitación de este emulador es la imposibilidad de ejecutar programas en código máquina, y la razón es evidente. El código máquina es un conjunto de instrucciones directas destinadas al microprocesador y escritas con el léxico definido por su fabricante. Por tanto, para que esta simulación fuera efectiva, el **64** debería disponer del mismo microprocesador, el **Z80;** pero el que tiene es el **6510,** de distintos fabricantes y juego de instrucciones.

En realidad están presentes casi todos los comandos del **Spectrum.** Hay excepciones como OUT, que está relacionado con posiciones de memoria del **Spectrum** y no sería coherente su utilización en el **64.**

Una nota curiosa. Sabido es que los **Microdrives** son una de las peculiaridades de **Sinclair** en sistemas de almacenamiento masivo, por tanto, no conectables al **64.** Pues bien, las unidades de *diskette* **1541** pasan a ser tratadas por el *software* como si de **Microdrives** se tratase. Los mensajes de error son también réplica de los empleados por el **Spectrum.**

Algunos datos de interés, proporcionados por el fabricante del programa, son: El emulador ocupa 25 Kbytes en la memoria del **64;** 10 Kbytes se destinan a la pantalla, y quedan 30 K libres para los programas en BASIC. La cifra es comparable con el modelo de 48 K y es superior al de 16 K. También se lamenta de que se vean mermadas las posibilidades del *chip* sintetizador de sonido, en favor del comando BEEP, mucho más modesto en sus pretensiones. Sin embargo, encuentra el consuelo de poder dibujar directamente líneas y círculos con la ayuda de DRAW y CIRCLE, algo que el **64** no hace espontáneamente.

Nuestra conclusión es de sorpresa ante la magnífica emulación de un ordenador tan diferente, en muchos aspectos, del **Commodore 64.** Incluso hemos llegado a sospechar si **Commodore** no habrá animado a esta gente a que les ayude, con su *software,* a «robarle» participación en el mercado al **Spectrum,** llevándose de calle a sus incondicionales.

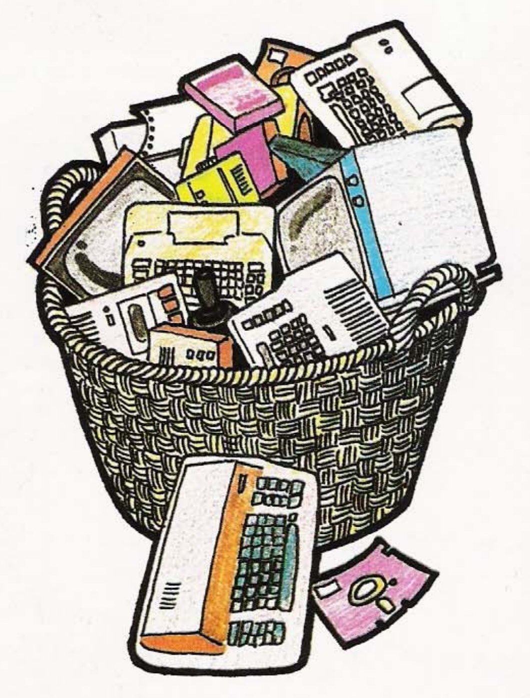
EL ZOCO DE INPUT

Todo se compra y se vende. Los antiguos zocos fueron lugares destinados a todo tipo de transacciones. INPUT también tiene el suyo. Vuestras operaciones de compra, cambio o venta serán publicadas en esta sección, pero dos son las limitaciones que imponemos:

a) La propuesta tendrá que ver con la microinformática.

b) Nos reservamos el derecho de no publicar aquellos insertos de los que se sospeche un trasfondo lucrativo.

Ahora un ruego. Tratar de resumir al máximo el texto; escribir casi como un telegrama siendo claros y concisos.



Envía tu mensaje a:

INPUT COMMODORE-ZOCO
c./ Alberto Alcocer, 46
28016 **MADRID**

FABRICANDO EXPLOSIONES

- CREACION DE FOGONAZOS EN LA PANTALLA
- BOMBARDEO AEREO
- COMO AÑADIR LLAMARADAS Y EXPLOSIONES

Teclea para C-64

Las explosiciones son un componente obligado del repertorio de los programadores de juegos, desde un combate aéreo hasta una guerra interestelar. Aquí veremos la manera de programar efectos visuales adecuados que podrás utilizar en una amplia variedad de juegos.

Es muy fácil conseguir que tus juegos sean mucho más espectaculares, añadiéndoles unas rutinas gráficas de efectos especiales. Y no tienen porqué ser programas muy complicados para que haya una gran diferencia en la apariencia.

Como verás, hay muchas maneras diferentes de producir efectos visuales; lo único que tienes que recordar es utilizar el tipo de efecto adecuado para cada tipo de juego.

Los efectos de llamarada que aquí describimos valen para toda clase de juegos en los que aparezcan edificios, coches, barcos, tanques, etc., pero no quedarían bien en un juego espacial.

Además, como las llamaradas y las explosiones se han construido usando GDUs, el máximo tamaño del blanco está limitado. Esto significa que no puedes incorporarlos sin más en cualquier juego sin modificar la rutina.

En el **Commodore 64** no es tan fácil programar efectos visuales como en otras máquinas domésticas. Las explosiones más convincentes se basan en gran medida en los *sprites* y en los gráficos de alta resolución de este ordenador. Esto significa que se requiere una larga rutina para su creación y, además, que normalmente se han de diseñar específicamente para el programa en que van a ser utilizados. Más adelante veremos en este mismo artículo un ejemplo de explosión generada con *sprites,* que puede utilizarse en juegos diferentes.

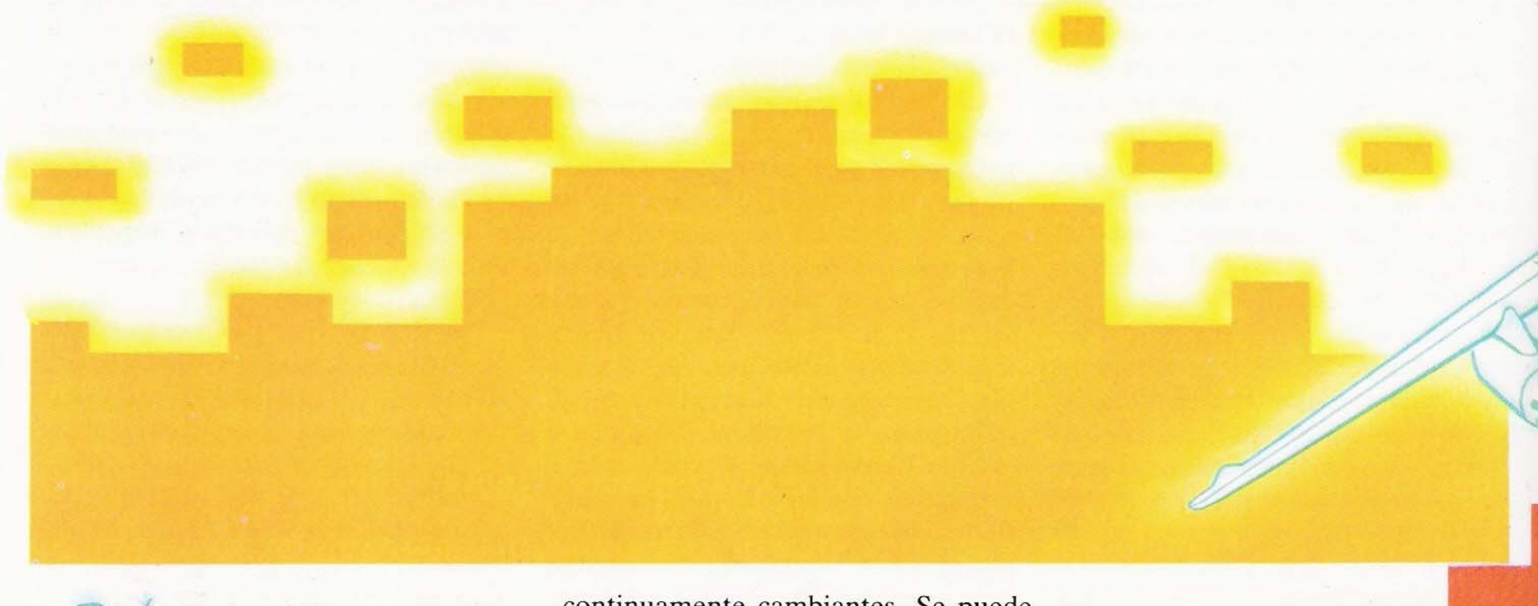
No obstante se pueden utilizar rutinas más sencillas, como ésta de barras intermitentes, que resultan muy eficaces en programas que requieren algún tipo de interrupción visual. La siguiente podría utilizarse para señalar el final de un juego o la pérdida de una «vida».

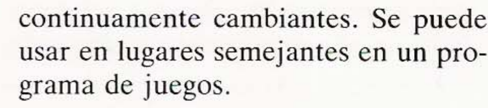
```
100 PRINT"♡"
```



```
110 FOR Z=0 TO 23:READ
    X:POKE 832+Z,X:NEXT Z
120 SYS 832:GOTO120
130 DATA 162,0,160,200,200,
    208,253,160,250,200,
    208,253,232
135 DATA 142,33,208,142,32,
    208,224,16,208,235,96
```

Una rutina más interesante es la que sigue a continuación, que presenta un diagrama de rectángulos de colores





continuamente cambiantes. Se puede usar en lugares semejantes en un programa de juegos.

```
5 C$="■≡±◣▒↑←π♠
  ◤⊠○♣○⊠◤♠π←↑
  ▒◣±≡■":PRINT "♡";
8 FOR Z=0 TO 48:READ
  X:POKE 49152+Z,X:NEXT Z
10 FOR Z=1 TO 24:PRINT MID$
  (C$,Z,1)"◘□□□□□□□
  □□□□□□□□□□□□□
  □□□□□□□□□□□□□
  □□□□□□□□□□□";
20 NEXT Z:PRINT "◙"
40 FOR Z=0 TO 12
60 FOR ZZ=55335-Z+(Z*40) TO
   56295-Z-(40*Z) STEP
   40:POKE ZZ,Z: NEXT ZZ
70 FOR ZZ=55296+Z+(Z*40) TO
   56256+Z-(40*Z) STEP
   40:POKE ZZ,Z: NEXT ZZ,Z
80 DATA 169,0,141,251,0,
   169,216,141,252,0,160,
   0,177,251,201,0,208,2,
   169,16,24
82 DATA 233,0,145,251,24,
   230,251,165,251,201,0,
```

```
    2Ø8,2,23Ø,252,165,251
84 DATA 2Ø1,232,2Ø8,224,
    165,252,2Ø1,219,2Ø8,
    218,98
100 SYS 49152:FOR Z=1 TO
    25Ø:NEXT Z:GOTO 1ØØ
```



La longitud de la rutina se controla con el bucle FOR...NEXT de la línea 100, que introduce un retardo de 250 unidades de tiempo. Este valor se puede ajustar al valor requerido, e incluso intentar suprimir completamente el bucle, para tener una representación que realmente canse a nuestra vista.

Eliminando las líneas 40, 60 y 70, perderás las barras verticales, quedando sólo bandas horizontales de color que se mueven hacia el centro. Esta forma puede ser preferible si quieres incluir algún tipo de mensaje en la pantalla, lo cual puedes hacer añadiendo tu comentario en la sentencia PRINT de la línea 20, después del símbolo del cursor.

```
2Ø NEXT Z:PRINT
   "■"TAB(2Ø)"HOLA"
```

Evidentemente, puedes encajar los caracteres en esta línea para componer tu mensaje adecuadamente.

EXPLOSION CON SPRITES

El breve programa que sigue es una sencilla demostración de la apariencia y funcionamiento de un programa de juegos típico, escrito en BASIC. Muestra un avión a reacción volando por la pantalla y bombardeando un edificio.

```
1Ø PRINT "♡":POKE
   5328Ø,Ø:POKE 53281,Ø
15 PRINT "■■■■■■■■■■■■■
   ■■■■■■■"TAB(2Ø)"■■◤
   ◣■▌▌▌▌π▒▬□■▌▌▌▌▌▌▌▌■
   □□□"
2Ø FOR Z=1 TO 2Ø:PRINT
   "■"TAB(Z)
   "□■▒◣■▬■▌▌▌▌□■□□"
3Ø POKE 1Ø24+Z*41,32:POKE
   1Ø65+Z*41,42:POKE
   55337+Z*41,1
4Ø NEXT Z:PRINT "■"TAB(Z)
   "□□■▌▌▌▌□□□□□"
```

Sin una «explosión», el efecto del bombardeo no resulta muy realista; intenta añadir pues estas líneas, que harán que el edificio se destruya lentamente.

```
45 FOR Z=1 TO 2Ø:POKE
   1884+RND(1)*2,RND(1)*15+1
   1Ø:NEXT Z
5Ø POKE 1885,223:POKE
   1845,32:POKE 1884,1Ø4
```

Incluso así, el efecto conseguido no es muy espectacular. Pero añádele ahora las siguientes líneas, que forman una rutina de explosión basada en los *sprites*.

```
  1 FOR Z=832 TO
    832+63*2:POKE Z,Ø:NEXT Z
  2 FOR Z=13 TO 14:FOR A=Ø TO
    15:FOR B=Ø TO 1: READ
    X:POKE Z*64+A*3+B,X:NEXT
    B,A,Z
  3 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,15,
    24Ø,15,24Ø,53,92,54,
    156,54,156,53,92,15,
    24Ø,15,24Ø
  4 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
  5 DATA 15,24Ø,15,24Ø,53,
    92,53,92,213,87,214,151,
    214,151,218,167,218,167
  6 DATA 214,151,214,151,
    213,87,53,92,53,92,15,
    24Ø,15,24Ø
9ØØ V=53248:POKE V+21,3:POKE
    V+32,Ø:POKE V+33,Ø:POKE
    V+37,2:POKE V+28,3
91Ø POKE V+27,3:A(Ø)=1:A(1)
    =8:B(Ø)=4:B(1)=7
915 FOR LL=1 TO 3:IF LL=1 OR
    LL=3 THEN POKE V+23,Ø:POKE
    V+29,Ø
917 POKE V,183:POKE V+1,
    216:DD=Ø
92Ø IF LL=2 THEN
    POKE V+23,3:POKE
    V+29,3:POKE V,175:POKE
    V+1,2Ø8:DD=2Ø
925 FOR UU=1 TO 1Ø:FOR Z=13
    TO 14:POKE 2Ø4Ø,Z:POKE
    V+39,B(RND(1)*2)
93Ø FOR T=1 TO DD:NEXT
    T:POKE V+38,A(RND(1)*2)
94Ø NEXT Z,UU,LL:POKE V+21,Ø
```

Ejecuta el programa (con RUN) para tener una explosión de colores. Esta rutina puede hacer cosas maravillosas incluso con programas de juegos sencillos.

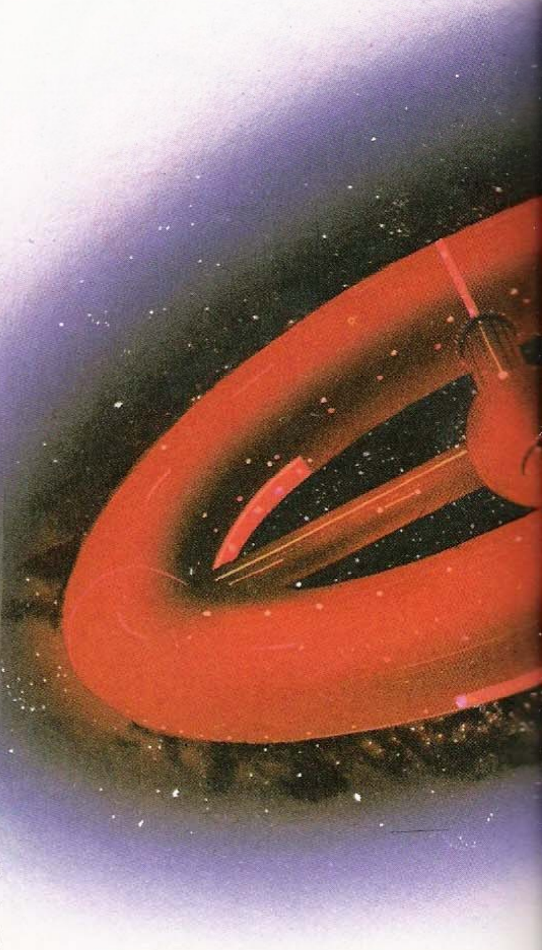
A pesar de su aparente complejidad, el programa es realmente muy sencillo; consiste básicamente en generar dos *sprites* y cambiar de uno a otro repetidamente (esto lo explicaremos con más detalla después). Los datos para la «explosión» a base de *sprites* están contenidos en cuatro líneas al principio del programa. Antes encontramos la rutina de inicialización, que primeramente borra un área de memoria (las direcciones 832 a 1022 realmente forman parte del *buffer* de Entrada/Salida de la cinta, pero están disponibles para que almacenemos los *sprites*). En la siguiente línea se leen los datos y se almacenan en las mismas direcciones, borradas. Observa que Z recibe los valores de 13 y 14, que al ser multiplicados por 64 en la línea 2, dan el comienzo del almacenamiento de cada uno, las direcciones 832 y 896. El número 64 refleja aquí la cantidad de memoria requerida por cada *sprite,* en total 63+1 bytes.

En todo programa, la rutina de inicialización ha de ir colocada al principio. Por eso la segunda parte del programa de explosión va numerada a partir de la línea 900, ya que normalmente será una subrutina que formará parte de un programa más largo.

La línea 900 hace operativos a los *sprites,* es decir los enciende, poniendo el color de la pantalla y el borde en negro (el valor 0 en la posición V+32 y V+33, en caso contrario se pueden alterar las direcciones 53280 y 53281), y cambiando el color del segundo «anillo» de cada *sprite* para realzar el efecto de la explosión. La última instrucción de esta línea habilita el modo multicolor para ambos *sprites.*

A partir de aquí el programa procede a activar el fondo, para que puedas ver los detalles a través de los sprites, coloreando el centro de la explosión. Puedes probar a cambiar los valores de los números que siguen a los signos igual de la línea 910, pero observa que sólo los colores «cálidos» y el blanco resultan efectivos.

La siguiente línea (915) controla el puntero, que señala al *sprite* pequeño o al grande, alternando entre ambos para conseguir el efecto pulsante de la explosión. Cuando los valores de LL son 1 y 3, está en uso el *sprite* pequeño, mientras que cuando el valor es 2, es el grande el activo. Se añade un efecto de parpadeo con el bucle FOR...NEXT al principio de la línea



930. La rutina de explosión termina en la línea 940, cuando se bloquea la posición de activación del *sprite* (53269).

La rutina de explosión se puede añadir a otros programas. Para mover la posición de los *sprites* tienes que ajustar los valores de V y V+1 en las líneas 915 y 920. V es la coordenada X de la pantalla, V+1 es la coordenada Y; la diferencia entre ambas es 33 en todos los casos. El valor de V en la línea 915 supera en 8 a cualquier valor que escojas para V en la línea 920. El valor de V+1 en las líneas 920 y 915 supera entonces en 33 a V.

ROS S

■ LAS RUTINAS PARA JUEGOS DE MARCIANITOS
■ DIBUJO DE LOS ELEMENTOS
■ INCORPORACION DEL MOVIMIENTO

Teclea para C-64



...asta los últi-...s, los enemi-...ndo siempre ...tienes la ma-...rarlos en una

...or aspecto si ...característi-...a resolución ...e servirte de ... Los progra-...solución se-... los que sólo ... del teclado, ...s resultados ...a.

...anitos están ...le enemigos ...s que dispa-...tenerse plá-...s aniquiles. ...mos un jue-go llamado Estación Espacial, que corre sobre el **Commodore 64** y te enseñará la manera de programar el movimiento aleatorio de un «invasor» por la pantalla, así como la forma de lanzar misiles contra un blanco, en este caso la estación espacial.

El jugador dispone de cuatro defensas o pantallas para protegerse de los misiles atacantes. Sin embargo no podrás usar todos los blindajes a la vez, ya que sólo tienes una cantidad de combustible limitada para moverlos.

A fin de que el juego resulte más difícil, el programa está diseñado de modo que el invasor se mueva de forma aleatoria, pudiendo también desaparecer en el hiperespacio y reaparecer de nuevo por otro sitio totalmente distinto.

Tal como se presente aquí el juego, no está realmente completo, ya que le falta la puntuación y el cronometraje.

Pero esto se remedia fácilmente con los métodos que presentamos en los capítulos anteriores.

Aunque el extraterrestre tiene la misma apariencia que en los juegos comerciales, la estación espacial es sólo una silueta. Si quieres pasar una media hora interesante, puedes volver a dibujar la tuya propia utilizando los GDUs (gráficos definidos por el usuario) en la forma indicada en varios artículos de INPUT. No obstante tienes

Sin una «explosión», el efecto del bombardeo no resulta muy realista; intenta añadir pues estas líneas, que harán que el edificio se destruya lentamente.

```
45 FOR Z=1 TO 2Ø:POKE
   1884+RND(1)*2,RND(1)*15+1
   1Ø:NEXT Z
5Ø POKE 1885,223:POKE
   1845,32:POKE 1884,1Ø4
```

Incluso así, el efecto conseguido no es muy espectacular. Pero añádele ahora las siguientes líneas, que forman una rutina de explosión basada en los *sprites*.

```
  1 FOR Z=832 TO
    832+63*2:POKE Z,Ø:NEXT Z
  2 FOR Z=13 TO 14:FOR A=Ø TO
    15:FOR B=Ø TO 1: READ
    X:POKE Z*64+A*3+B,X:NEXT
    B,A,Z
  3 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,15,
    24Ø,15,24Ø,53,92,54,
    156,54,156,53,92,15,
    24Ø,15,24Ø
  4 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
  5 DATA 15,24Ø,15,24Ø,53,
    92,53,92,213,87,214,151,
    214,151,218,167,218,167
  6 DATA 214,151,214,151,
    213,87,53,92,53,92,15,
    24Ø,15,24Ø
9ØØ V=53248:POKE V+21,3:POKE
    V+32,Ø:POKE V+33,Ø:POKE
    V+37,2:POKE V+28,3
91Ø POKE V+27,3:A(Ø)=1:A(1)
    =8:B(Ø)=4:B(1)=7
915 FOR LL=1 TO 3:IF LL=1 OR
    LL=3 THEN POKE V+23,Ø:POKE
    V+29,Ø
917 POKE V,183:POKE V+1,
    216:DD=Ø
92Ø IF LL=2 THEN
    POKE V+23,3:POKE
    V+29,3:POKE V,175:POKE
    V+1,2Ø8:DD=2Ø
925 FOR UU=1 TO 1Ø:FOR Z=13
    TO 14:POKE 2Ø4Ø,Z:POKE
    V+39,B(RND(1)*2)
93Ø FOR T=1 TO DD:NEXT
    T:POKE V+38,A(RND(1)*2)
94Ø NEXT Z,UU,LL:POKE V+21,Ø
```

Ejecuta el
para tener u
Esta rutina p
llosas incluso
gos sencillos.

A pesar d
dad, el prog
sencillo; cons
nerar dos *sp*
otro repetida
mos con más
tos para la «e
tes están cont
principio del
tramos la ruti
primeramente
moria (las dir
mente forma
trada/Salida d
ponibles par
sprites). En l
los datos y s
mas direccio
que Z recibe
que al ser mu
nea 2, dan el
miento de c
832 y 896. E
la cantidad d
cada *sprite*, e

En todo pr
cialización ha
pio. Por eso
grama de ex
partir de la lí
mente será u
rá parte de u

La línea 9
sprites, es de
do el color de
negro (el valor ... V+32 y V+33, en caso contrario se pueden alterar las direcciones 53280 y 53281), y cambiando el color del segundo «anillo» de cada *sprite* para realzar el efecto de la explosión. La última instrucción de esta línea habilita el modo multicolor para ambos *sprites*.

A partir de aquí el programa procede a activar el fondo, para que puedas ver los detalles a través de los sprites, coloreando el centro de la explosión. Puedes probar a cambiar los valores de los números que siguen a los signos igual de la línea 910, pero observa que sólo los colores «cálidos» y el blanco resultan efectivos.

930. La rutina de explosión termina en la línea 940, cuando se bloquea la posición de activación del *sprite* (53269).

La rutina de explosión se puede añadir a otros programas. Para mover la posición de los *sprites* tienes que ajustar los valores de V y V+1 en las líneas 915 y 920. V es la coordenada X de la pantalla, V+1 es la coordenada Y; la diferencia entre ambas es 33 en todos los casos. El valor de V en la línea 915 supera en 8 a cualquier valor que escojas para V en la línea 920. El valor de V+1 en las líneas 920 y 915 supera entonces en 33 a V.

ROS
S

- LAS RUTINAS PARA JUEGOS DE MARCIANITOS
- DIBUJO DE LOS ELEMENTOS
- INCORPORACION DEL MOVIMIENTO

Teclea para C-64

**asta los últi-
s, los enemi-
ndo siempre
tienes la ma-
rarlos en una**

jor aspecto si
característi-
ta resolución
le servirte de
Los progra-
resolución se-
los que sólo
del teclado,
los resultados
na.
cianitos están
de enemigos
es que dispa-
letenerse plá-
los aniquiles.
amos un jue-

go llamado Estación Espacial, que corre sobre el **Commodore 64** y te enseñará la manera de programar el movimiento aleatorio de un «invasor» por la pantalla, así como la forma de lanzar misiles contra un blanco, en este caso la estación espacial.

El jugador dispone de cuatro defensas o pantallas para protegerse de los misiles atacantes. Sin embargo no podrás usar todos los blindajes a la vez, ya que sólo tienes una cantidad de combustible limitada para moverlos.

A fin de que el juego resulte más difícil, el programa está diseñado de modo que el invasor se mueva de forma aleatoria, pudiendo también desaparecer en el hiperespacio y reaparecer de nuevo por otro sitio totalmente distinto.

Tal como se presente aquí el juego, no está realmente completo, ya que le falta la puntuación y el cronometraje.

Pero esto se remedia fácilmente con los métodos que presentamos en los capítulos anteriores.

Aunque el extraterrestre tiene la misma apariencia que en los juegos comerciales, la estación espacial es sólo una silueta. Si quieres pasar una media hora interesante, puedes volver a dibujar la tuya propia utilizando los GDUs (gráficos definidos por el usuario) en la forma indicada en varios artículos de INPUT. No obstante tienes

Sin una «explosión», el efecto del bombardeo no resulta muy realista; intenta añadir pues estas líneas, que harán que el edificio se destruya lentamente.

```
45 FOR Z=1 TO 2Ø:POKE
   1884+RND(1)*2,RND(1)*15+1
   1Ø:NEXT Z
5Ø POKE 1885,223:POKE
   1845,32:POKE 1884,1Ø4
```

Incluso así, el efecto conseguido no es muy espectacular. Pero añádele ahora las siguientes líneas, que forman una rutina de explosión basada en los *sprites*.

```
  1 FOR Z=832 TO
    832+63*2:POKE Z,Ø:NEXT Z
  2 FOR Z=13 TO 14:FOR A=Ø TO
    15:FOR B=Ø TO 1: READ
    X:POKE Z*64+A*3+B,X:NEXT
    B,A,Z
  3 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,15,
    24Ø,15,24Ø,53,92,54,
    156,54,156,53,92,15,
    24Ø,15,24Ø
  4 DATA Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
  5 DATA 15,24Ø,15,24Ø,53,
    92,53,92,213,87,214,151,
    214,151,218,167,218,167
  6 DATA 214,151,214,151,
    213,87,53,92,53,92,15,
    24Ø,15,24Ø
9ØØ V=53248:POKE V+21,3:POKE
    V+32,Ø:POKE V+33,Ø:POKE
    V+37,2:POKE V+28,3
91Ø POKE V+27,3:A(Ø)=1:A(1)
    =8:B(Ø)=4:B(1)=7
915 FOR LL=1 TO 3:IF LL=1 OR
    LL=3 THEN POKE V+23,Ø:POKE
    V+29,Ø
917 POKE V,183:POKE V+1,
    216:DD=Ø

92Ø IF LL=2 THEN
    POKE V+23,3:POKE
    V+29,3:POKE V,175:POKE
    V+1,2Ø8:DD=2Ø
925 FOR UU=1 TO 1Ø:FOR Z=13
    TO 14:POKE 2Ø4Ø,Z:POKE
    V+39,B(RND(1)*2)
93Ø FOR T=1 TO DD:NEXT
    T:POKE V+38,A(RND(1)*2)
94Ø NEXT Z,UU,LL:POKE V+21,Ø
```

Ejecuta el
para tener u
Esta rutina p
llosas incluso
gos sencillos.
A pesar d
dad, el prog
sencillo; cons
nerar dos *spr*
otro repetida
mos con más
tos para la «e
tes están cont
principio del
tramos la ruti
primeramente
moria (las dir
mente forman
trada/Salida d
ponibles para
sprites). En l
los datos y se
mas direccion
que Z recibe
que al ser mul
nea 2, dan el
miento de ca
832 y 896. El
la cantidad de
cada *sprite*, e
En todo pr
cialización ha
pio. Por eso l
grama de ex
partir de la lí
mente será u
rá parte de u
La línea 90
sprites, es dec
do el color de
negro (el valo
y V+33, en caso contrario se pueden alterar las direcciones 53280 y 53281), y cambiando el color del segundo «anillo» de cada *sprite* para realzar el efecto de la explosión. La última instrucción de esta línea habilita el modo multicolor para ambos *sprites*.

A partir de aquí el programa procede a activar el fondo, para que puedas ver los detalles a través de los sprites, coloreando el centro de la explosión. Puedes probar a cambiar los valores de los números que siguen a los signos igual de la línea 910, pero observa que sólo los colores «cálidos» y el blanco resultan efectivos.

930. La rutina de explosión termina en la línea 940, cuando se bloquea la posición de activación del *sprite* (53269).

La rutina de explosión se puede añadir a otros programas. Para mover la posición de los *sprites* tienes que ajustar los valores de V y V+1 en las líneas 915 y 920. V es la coordenada X de la pantalla, V+1 es la coordenada Y; la diferencia entre ambas es 33 en todos los casos. El valor de V en la línea 915 supera en 8 a cualquier valor que escojas para V en la línea 920. El valor de V+1 en las líneas 920 y 915 supera entonces en 33 a V.

ENEMIGOS MORTIFEROS Y EXTRATERRESTRES

- LAS RUTINAS PARA JUEGOS DE MARCIANITOS
- DIBUJO DE LOS ELEMENTOS
- INCORPORACION DEL MOVIMIENTO

Teclea para C-64



Desde Los Invasores, hasta los últimos juegos de marcianitos, los enemigos que atacan disparando siempre han sido un desafío. Aquí tienes la manera de crearlos e incorporarlos en una rutina completa de juego.

Los juegos tendrán mejor aspecto si utilizamos algunas de las características de los gráficos de alta resolución de tu máquina, en lugar de servirte de los caracteres ordinarios. Los programas con gráficos de alta resolución serán más complicados que los que sólo emplean a los caracteres del teclado, pero ten por seguro que los resultados realmente merecen la pena.

Muchos juegos de marcianitos están basados en la presencia de enemigos invasores o extraterrestres que disparan contra tí, en vez de detenerse plácidamente a esperar que los aniquiles. Seguidamente te presentamos un juego llamado Estación Espacial, que corre sobre el **Commodore 64** y te enseñará la manera de programar el movimiento aleatorio de un «invasor» por la pantalla, así como la forma de lanzar misiles contra un blanco, en este caso la estación espacial.

El jugador dispone de cuatro defensas o pantallas para protegerse de los misiles atacantes. Sin embargo no podrás usar todos los blindajes a la vez, ya que sólo tienes una cantidad de combustible limitada para moverlos.

A fin de que el juego resulte más difícil, el programa está diseñado de modo que el invasor se mueva de forma aleatoria, pudiendo también desaparecer en el hiperespacio y reaparecer de nuevo por otro sitio totalmente distinto.

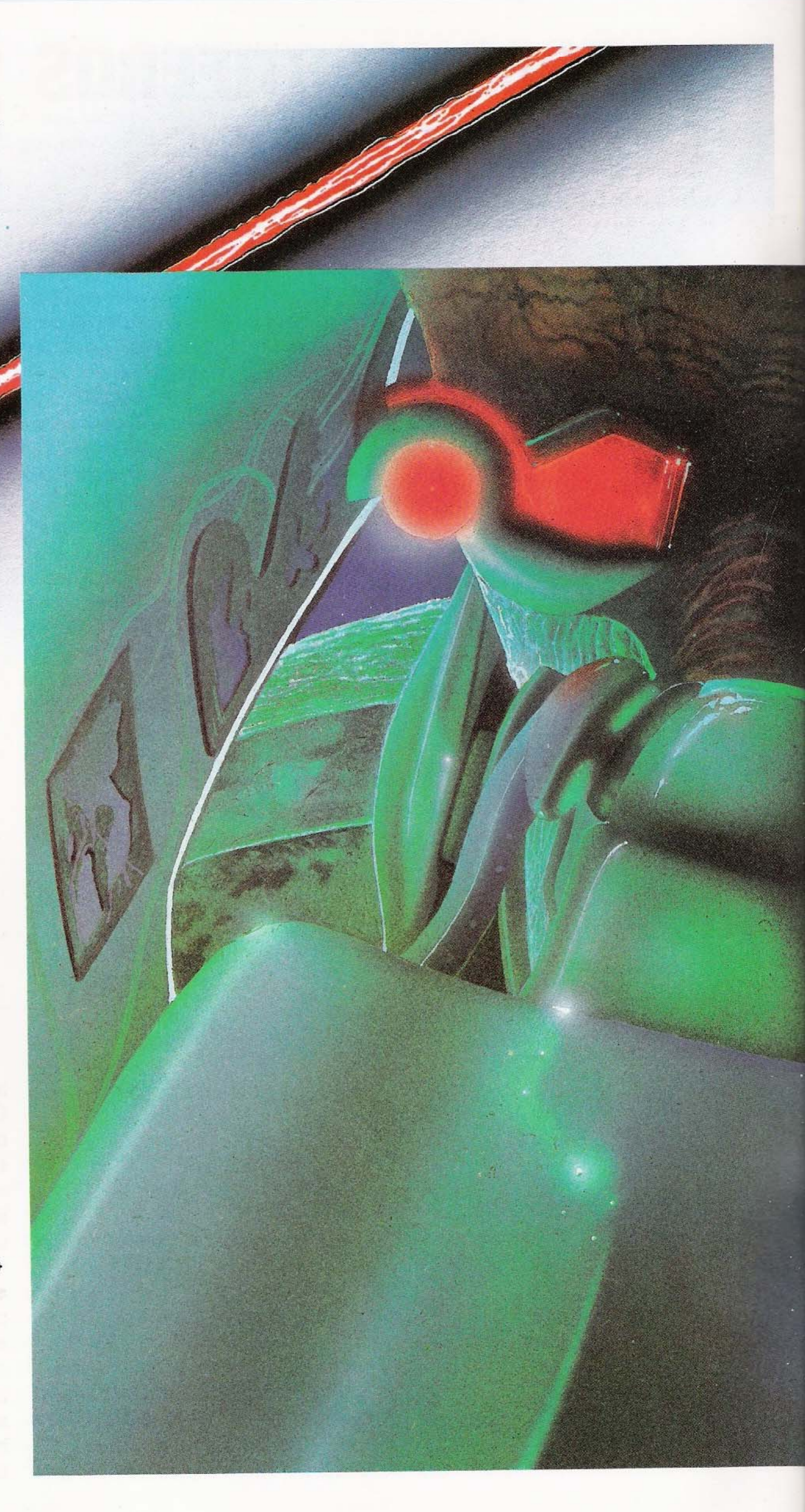
Tal como se presente aquí el juego, no está realmente completo, ya que le falta la puntuación y el cronometraje.

Pero esto se remedia fácilmente con los métodos que presentamos en los capítulos anteriores.

Aunque el extraterrestre tiene la misma apariencia que en los juegos comerciales, la estación espacial es sólo una silueta. Si quieres pasar una media hora interesante, puedes volver a dibujar la tuya propia utilizando los GDUs (gráficos definidos por el usuario) en la forma indicada en varios artículos de INPUT. No obstante tienes

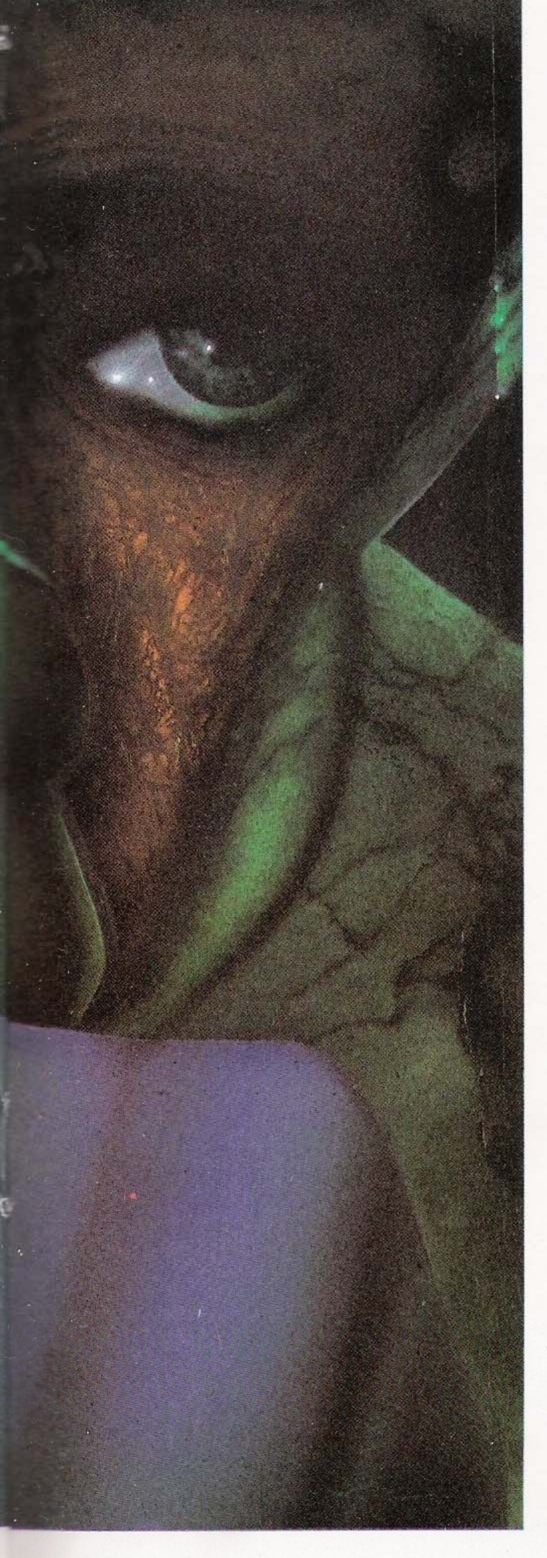
que tener cuidado de mantenerte dentro del área utilizada por nuestra estación espacial. De no hacerlo así, podría solaparse con las pantallas defensivas, requiriendo una revisión demasiado laboriosa.

La presente versión del juego de la estación espacial utiliza los *sprites*, cuya información está contenida en un gran número de sentencias DATA cerca del comienzo del programa.



```
10 POKE 56,100:POKE
   55,0:POKE 52,100:POKE
   51,0:CLR
20 DATA 0,254,0,3,57,128,7,
   255,192,0,16,0,16,56,
   16,56,84
25 DATA 56,124,146,124,
   131,255
30 DATA 130,144,56,18,184,
   16,58,144,16,18,131,
   255,130,254,84,254,252
35 DATA 56,126,0,56
40 DATA 0,0,40,0,0,56,0,1,
   199,0,6,16,192,1,199,
   0,0,124,0
50 DATA 0,0,0,127,255,254,
   64,0,2,64,0,2,64,16,
   2,64,48,2,64,16,2,64,
   16,2,80,56
60 DATA 114,84,0,18,94,0,
   114,68,0,66,68,56,114,
   64,8,2,64,56,2,64,8,2,64
70 DATA 56,2,64,0,2,64,0,
   2,127,255,254,0,0,0
80 DATA 0,0,0,0,0,0,0,0,0,
   0,0,0,0,0,0,0,0,0,0,0,
   0,0,8,0,0,42,0,0,20,
   0,0,42,0,0
90 DATA 42,0,0,73,0,0,8,0,
   0,0,0,0,0,0,0,0,0,0,0,
```

```
    0,0,0,0,0,0,0,0,0,0
100 DATA 0,0,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0,
    0,0,0,0,0,3,3,0,4,132,
    128,255
110 DATA 255,255,72,72,68,
    48,48,56,0,0,0,0,0,0,
    0,0,0,0,0,0,0,0,0,0,
```



```
    0,0,0,0,0,0
120 DATA 0,0,16,0,0,24,0,
    0,20,0,0,20,0,0,20,0,0,
    24,0,0,48
125 DATA 0,0,80,0,0,80,0,0,
    48,0
130 DATA 0,24,0,0,20,0,0,
    20,0,0,24,0,0,48,0,0,
    80,0,0,80,0,0,80,0,0,
    48,0,0,16
140 DATA 0,0,16,0
210 FOR ZZ=0 TO 4:POKE
    2040+ZZ, 200+ZZ
220 FOR Z=1 TO 63:READ X:POKE
    12799+(ZZ*64)+Z,X:NEXT
    Z,ZZ
230 CLR:V=53248:FU=100:POKE
    650,255:POKE 53280,0:
    POKE 53281,0: PRINT "♥"
240 POKE V+2,145:POKE
    V+3,120: POKE
    V+23,250:POKE V+29,250:
    POKE V+30,240
250 XX=31+INT(RND(1)*210):YY
    =60:DX=1:DY=1:IF
    RND(1)>.50 THEN
    YY=180:DY=-DY
260 PRINT "■"TAB(14)"π FUEL:
    □□□□▮▮▮▮▮▮▮▮" FU
270 IF RND(1)>.90 THEN XX=
    31+IN T(RND(1)*210):YY=
    60:DX=1:DY=1
320 XX=XX+DX:IF XX=<30 OR
    XX=>245 THEN DX=-DX
330 YY=YY+DY:IF YY=<50 OR
    YY=>190 THEN DY=-DY
340 POKE V,XX:POKE V+1,YY:IF
    F=0 THEN F=1:FX=XX:FY=YY
350 IF F=1 THEN GOSUB 410
360 GET A$:S$="":SH=0
370 IF A$="4" OR A$="2" THEN
    SH=1:POKE 2043,204:
    S$=A$
380 IF A$="1" OR A$="3" THEN
    SH=1:POKE 2043,203:
    S$=A$
390 IF SH=1 THEN 470
400 POKE V+21,247:GOTO 260
410 IF FX>153 THEN FX=FX-5
420 IF FX<153 THEN FX=FX+5
430 IF FY<135 THEN FY=FY+5
440 IF FY>135 THEN FY=FY-5
450 POKE V+4,FX:POKE
    V+5,FY:IF PEEK
    (V+30)=246 THEN 550
460 RETURN
470 IF S$="4" THEN L1=118:
    L2=120
480 IF S$="2" THEN L1=175:
    L2=120
490 IF S$="1" THEN L1=145:
    L2=95
500 IF S$="3" THEN L1=145:
    L2=145
510 POKE V+6,L1:POKE
    V+7,L2:FU=FU-1:POKE
    V+21,255:IF FU<0THEN 540
520 IF PEEK(V+30)=252 THEN
    F=0:GOTO 240
530 GOTO 260
540 PRINT TAB(4)"■■ESTAS
    SIN FUEL!":GOTO 560
550 PRINT TAB(11)"■■TE HAN
    DADO!"
560 FOR ZZ=1 TO 10:FOR
    T=1 TO 100:NEXT:POKE
    V+21,247
565 FOR T=1 TO 100:NEXT:POKE
    V+21,0
570 NEXT:PRINT "♥"TAB(6)
    "■■■■■■■■■■■■■■
    PRESIONA LA BARRA
    DE ESPACIOS"
580 GET X$:IF X$<>"□" THEN
    580
590 RUN 230
```

La primera línea del programa reserva espacio para el *sprite* en la memoria del ordenador, de modo que el programa BASIC utilizado para moverlo no perturbe al programa del propio *sprite*.

Las líneas 20 a 140 contienen la información necesaria para los *sprites* de la estación espacial, defensas, misiles y extraterrestres. Las líneas 210 y 220 sitúan los **punteros** de los *sprites* y almacenan sus datos en la memoria. Cada uno ocupa 64 bytes (3 veces 21, más uno extra), habiendo cinco de ellos, lo cual explica el significado de estos valores en estas dos líneas de programa.

Las siguientes dos líneas inicializan el ordenador, ajustan las variables, las diversas posiciones de los sprites, la autorrepetición de las teclas y los colores.

El programa continúa en la línea 250, creando una posición aleatoria para el *sprite* invasor, que aparece sobre una línea de la parte superior de la pantalla o una línea del fondo, de-



alcanzar el centro de la estación espacial sin ser interrumpido —o lo que es lo mismo, si en la línea 520 no se ha detectado la colisión de un *sprite*— se satisface la condición de la línea 450. Comienza entonces la rutina de fin de juego en las líneas 540 y siguientes, presentándote un mensaje y poniendo la pantalla con intermitencia antes de ofrecerte otra partida.

Cada vez que se activa la pantalla, se disminuye el contador de carburante, la variable FU. Cada vez que el programa vuelve a este punto, los con-

pendiendo del valor obtenido por la función RND.

El programa empieza en la línea 260 con la pantalla en blanco, el combustible que queda y el invasor. Este lanza un misil y sale corriendo. Las líneas 360 a 400 activan las defensas de la estación espacial dependiendo de cuál de las teclas 1, 2, 3 ó 4 haya sido pulsada. Si se ha activado alguna de las defensas, se cumple la condición de la línea 390 y el programa salta a la rutina de las líneas 470 a 530, que controla la posición del *sprite* de la defensa activada.

Con la rutina que abarca las líneas 410 a 460, el misil se aproxima a su blanco. Si el *sprite* del misil consigue

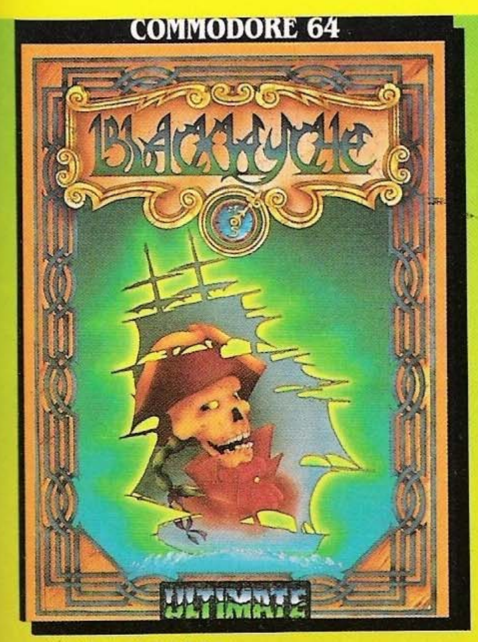
troles de cursor del indicador de combustible de la sentencia PRINT de la línea 260 mueven el cursor un lugar hacia la izquierda, con lo que de hecho se elimina de la anterior cantidad de fuel antes de añadir el nuevo valor de FU. Cuando el valor de FU es menor que cero en la línea 510, se produce un salto a la parte principal de la rutina de fin de juego.

¡NUEVO!

SIEMPRE LOS PRIMEROS EN TENER LO ULTIMO



P.º de la Castellana, 268, 3.º C. 28046-MADRID.
Tel.: (91) 733 25 00



BLACKWYCH

La última novedad de ULTIMATE para Commodore. Vuelven las aventuras del héroe de Staff of Karnath y Entombed, Sir Arthur Pendragon, en un juego todavia, si cabe, más apasionante.

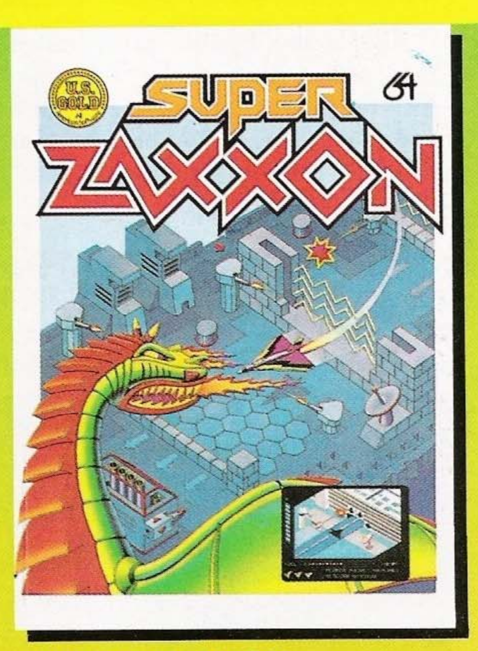
P.V.P.: 2.100 ptas.
Precio Socios C. de Soft: 1.890 ptas.



BEACH HEAD II

Ya está aqui la esperada continuación de Beach Head, el juego más vendido para Commodore. Los gráficos y el nuevo simulador de voz que se incluyen en el Beach Head II te harán sentirte en pleno centro de la batalla por la conquista de la fortaleza enemiga.

P.V.P.: 2.900 ptas.
Precio Socios C. de Soft: 2.600 ptas.



SUPER ZAXXON

La continuación de Zaxxon supera a la primera no solamente en gráficos y realismo, sino que la imaginación con que está desarrollado el juego alcanza niveles sencillamente extraordinarios.

P.V.P.: 2.600 ptas.
Precio Socios C. de Soft: 2.340 ptas.

¡¡¡...Y LOS TRES PROGRAMAS POR SOLO 6.300 PTAS.!!!

¡HAZTE HOY MISMO SOCIO DEL CIRCULO DE SOFT! Además de poder adquirir tus programas al mejor precio, recibirás información de forma periódica y gratuita, del mejor software que aparezca en el mercado.

¿QUE HAY QUE HACER PARA SER SOCIO DEL CIRCULO DE SOFT? Asi de fácil: envíanos por correo tu nombre, dirección y modelo de ordenador, o bien, pide por teléfono o por correo tu primer programa. ¡Y entrarás a formar parte del CIRCULO DE SOFT de forma inmediata!

☐ SÍ, quiero ser SOCIO desde hoy mismo del CIRCULO DE SOFT y recibir periódicamente información de novedades de software, así como beneficiarme desde hoy mismo de los precios reducidos reservados a los SOCIOS y de sus Ofertas Especiales. El ser SOCIO no me obliga a compra alguna.

Si prefieres formalizar tu compra por teléfono puedes hacerlo llamando al (91) 733 25 00. ¡¡NO SE COBRAN LOS GASTOS DE ENVIO POR CORREO!!

TITULO	P.V.P.	ORDENADOR

☐ Contrarreembolso ☐ Giro Postal ☐ Talón adjunto a Microamigo, S.A. ☐ Tarjeta VISA n.º ________ Fecha caducidad ________

Nombre ________ Apellidos ________ Edad ________

Domicilio ________ Teléfono ________

Localidad ________ C.P. ________ Provincia ________

ENVIA MENSAJES SECRETOS (y II)

- RUPTURA DE CODIGOS
- FRECUENCIA
- NUEVOS METODOS DE CODIFICACION
- DICCIONARIOS DE CODIGOS

Aquí tienes algunos procedimientos más sutiles para enviar información secreta, preservándola de caer en manos inadecuadas. También se examinan brevemente algunos métodos de ruptura de códigos.

Como vimos en el artículo anterior (en nuestro número de septiembre) el problema de la codificación de la información reservada admite muchos planteamientos. Algunos de estos métodos son relativamente fáciles de descifrar, pero con ayuda de los ordenadores es posible emplear procedimientos cada vez más complicados.

RUPTURA DE CODIGOS

A medida que los criptógrafos se esfuerzan en inventar códigos mejores y más seguros se produce un esfuerzo paralelo por parte de los «rompedores de códigos», que hacen todo lo posible por frustar el esfuerzo de los primeros.

En los casos más sencillos de cifrado por **transposición** o **sustitución** se dispone de una potente herramienta, consistente en contar la frecuencia con la que aparecen todas las letras.

Cada idioma dispone de una serie de letras que aparecen con más frecuencia en un texto cualquiera que otras, en castelleno éste es el caso de las E, A, O, N, S, I. Según esto, si después de contar el número de veces que aparece cada letra en un texto cifrado siguen siendo las más frecuentes EAONSI, lo más probable es que te encuentres ante un cifrado por **transposición.** Si, por el contrario, son otras las letras que aparecen con mayor frecuencia, deberás considerar la posibilidad de que se trata de un cifrado por **sustitución.**

En ambos resulta necesario contar la frecuencia de cada letra en el texto del mensaje. Esta es una labor que manualmente ejecutada requiere tiempo y está sujeta a errores. El programa de distribución de la frecuencia que te ofrecemos a continuación te ahorra ese tedioso trabajo. Todo lo que tienes que hacer es introducir el texto, sin dejar espacios en blanco entre palabras, y cuando hayas terminado presiona RETURN. Volverá a aparecer la leyenda de introducción del texto y esta vez teclearemos únicamente un asterisco y RETURN. De esta forma, el ordenador presenta una lista de las letras del alfabeto y la cantidad de veces que ha sido utilizada cada una. Como observación, cabe decir que cuando escribimos el texto hay unos instantes de aparente inactividad por parte del ordenador, ocupado en la tarea de procesamiento del mensaje.

A continuación teclea el programa y observa cómo funciona todo ello en la práctica.

Teclea para C-64 y Vic-20

```
30 PRINT"♥■>■■CONTEO DE
   FRECUENCIA"
50 PRINT"■■NO DEJES
   ESPACIOS":PRINT"ENTRE
   PALABRAS"
60 DIM N(28)
70 PRINT"■PARA TERMINAR LA
   ENTRADA DE TEXTO Y"
80 FOR T=1 TO
   28:N(T)=0:NEXT T
85 PRINT"OBTENER EL
   RESULTADO TECLEA *"
90 INPUT"■ESCRIBE EL
   TEXTO♥";A$
100 IF A$="*"THEN 180
110 PRINT"♥"
120 FOR I=1 TO LEN(A$)
130 FOR J=1 TO 26
140 IF J=ASC(MID$(A$,I,1))-
    64 THEN N(J)=N(J)+1
150 NEXT J
160 NEXT I
170 GOTO 90
180 PRINT"♥"
190 PRINT"■LETRA■FREC■■■
    LETRA■FREC■"
200 FOR I=1 TO 13
210 PRINT CHR$(64+1);TAB(5);
    N(I);TAB(13);CHR$(77+I);
    TAB(18);N(13+I)
220 NEXT I
```



En cuanto a su funcionamiento, el programa se limita a ejecutar un procedimiento de conteo. En la primera fase del mismo asigna el valor 0 a cada una de las 28 variables indexadas. En dichas variables se almacenan los resultados parciales y totales de las 26 letras. Las dos últimas se incluyen para el caso de que quiera modificar el programa para obtener porcentajes a cualquier otro dato estadístico aislado.

CODIGO DE MULTIPLICACION

Durante la guerra civil norteamericana se utilizó el código de la empalizada para enviar mensajes secretos. Su funcionamiento es más o menos el siguiente. Supongamos que quieres enviar este mensaje: CONSULTA VENTA URGENTE, que podría tener un interés comercial. En una hoja de papel cuadriculado escribe la primera letra del mensaje en la línea superior, la segunda letra en la segunda línea, la tercera en la primera línea y así, sucesivamente con las demás, hasta obtener:

CNUTVNARET
OSLAETUGNE

que, como verás, recuerda a los postes ferroviarios colocados al tresbolillo. Este mensaje se puede codificar como CNUTVNARET OSLAETUGNE o también puede trocearse en longitudes de palabra más reales, así:

CNUTV NARETOS LAETUGNE

El código de la empalizada es realmente un caso especial de los modernos códigos multiplicativos. Ignorando los espacios, el texto original del mensaje en claro contiene 20 caracteres. Estos pueden ser almacenados en cadenas de 2 × 10, 10 × 2, 5 × 4 ó 4 × 5 elementos, tal como hemos visto.

Todo el que intente romper el código sin un conocimiento previo de la clave de multiplicación tendrá, sin duda, gran cantidad de trabajo ante sí.



Lo que ocurrirá es que iremos llenando la página con el mensaje en sentido vertical, hasta llenar la primera cadena. Seguidamente se comienza con el principio de la segunda y así sucesivamente. Una vez todas llenas podemos descifrar el mensaje. Es por esto que resulta una buena idea el hecho de intercalar algunas letras falsas en el mensaje para dificultar el trabajo de los intrusos.

El programa utiliza también la función de troceo y manipulación de cadenas (entre las líneas 140 a 210). El código de multiplicación simplemente lee el mensaje de una cadena en una dirección y lo visualiza en la otra. Sencillo, pero efectivo.

Teclea para Commodore-64

```
 30 PRINT"◘■>■■CODIGO DE
    MULTIPLICACION"
 50 PRINT"■■■NO DEJES
    ESPACIOS ENTRE PALABRAS"
 60 INPUT"■TEXTO■";M$
 70 INPUT"FILAS■";M
 80 INPUT"COLUMNAS■";N
 90 INPUT"CODIFICA(C) O
    DECODIFICA(D)";E$
100 PRINT"◘"
110 IF E$="C" THEN X=M
120 IF E$="D" THEN X=N
130 DIM D$(X)
140 FOR I=1 TO X
150 D$(I)=""
160 FOR J=1 TO LEN(M$) STEP X
170 B$=MID$(M$,I+J-1,1)
180 D$(I)=D$(I)+B$
190 NEXT J
200 PRINT D$(I);
210 NEXT I
```

LIBRO DE CLAVES

Hasta ahora solamente hemos considerado los métodos de cifrado. Las claves, propiamente dicho, son palabras o frases completas cifradas por otras palabras o números. Son tradicionalmente empleadas por grandes empresas que operan partiendo de premisas fijas. Las embajadas, los barcos y los negocios caen en esta categoría. Operando desde ubicaciones fijas resulta preferible, porque se precisan uno o más gruesos diccionarios para traducir el texto en bruto.

El programa del libro de claves dispone de un pequeño diccionario de 20 palabras. Lo primero que hace el programa es leer (READ) los datos incorporados en las sentencias DATA, que son las palabras y el número del código que le hacemos corresponder a cada una. Simultáneamente, las va depositando en la cadena bidimensional AS(I,J).

La siguiente sección (líneas 120 a 170) toma una palabra del mensaje y proporciona el número asociado como respuesta. Si, por ejemplo, AS(1,6) = SALIDA DESDE obtendremos en la pantalla el contenido del elemento AS(2,6) = «68677».

Un mensaje más largo, que podría ser: SALIDA DESDE NUEVA YORK A MEDIANOCHE. LLEGADA A PARIS DOMINGO MEDIANOCHE, será codificado como: 68677 54982 10996 26569 68719 10996 90075 48553 26569.

Por otro lado, el mensaje en clave: 74891 22317 10996 23874 se traduce como: ENVIA DINERO A ROMA.

Con este sencillo programa será más rápido codificar y decodificar que empleando el método manual, pues cuando las palabras deben ser elegidas entre cientos o miles el ahorro de esfuerzo es evidente.

Teclea para C-64 y Vic-20

```
 30 PRINT"◘■>■■LIBRO DE
    CODIGOS"
 50 DIM A$(2,20)
 60 FOR I=1 TO 2
 70 FOR J=1 TO 20
 80 READ A$(I,J)
 90 NEXT J,I
100 PRINT"■■QUIERES
    CODIFICAR(0) O
    DECODIFICAR(1)":INPUT X
120 INPUT"■ESCRIBE LA
    PALABRA■;M$
130 IF M$="*" THEN 280
140 FOR T=1 TO 20
150 IF M$=A$(1+X,T) THEN
    PRINT A$(2-X,T)
160 NEXT T
170 GOTO 120
180 DATA NUEVA YORK,
    LONDRES,PARIS,ROMA
190 DATA LLEGADA,SALIDA
    DESDE,VA
    A,ESCAPA,SABADO
200 DATA DOMINGO,TARDE,
    AMANECER,MEDIANOCHE
210 DATA NOCHE,A,EN,EL,
    ENVIA
220 DATA DINERO,COMIDA
230 DATA 54982,73581,90075,
    23874
240 DATA 68719,68677,10327,
    40476
250 DATA 27921,48553,11072,
    70355
260 DATA 26569,74832,10996,
    12128
270 DATA 69783,74891,22317,
    98724
280 END
```



Dos importantes métodos de ruptura de códigos.



De todas formas es posible aumentar el número de palabras a más de 20, basta con variar las cifras que van entre paréntesis detrás de DIM AS, aumentar la cantidad de palabras y códigos en las sentencias DATA y alterar los bucles FOR I y FOR J, para que ejecuten la lectura de datos las veces que sea preciso.

GRAFICOS EN ALTA RESOLUCION PARA EL COMMODORE 64 (II)

El primer capítulo de esta serie terminaba con una rutina misteriosa. Encontrando el símbolo @ será capaz de reconocer que está ante una de las nuevas palabras clave **no disponibles inicialmente en el ordenador.**

Pasemos al análisis de la rutina en cuestión. En primer lugar, el registro X es puesto a cero y se compara el contenido del acumulador (A) con las letras contenidas en una tabla, que comienza en la dirección $CF00. Cuando encuentra una coincidencia el procesador bifurca hacia el principio de la siguiente rutina, situada en la dirección $C061.

Si, por el contrario, no encuentra la coincidencia, se compara el número $0D con el byte procedente de la tabla. Ese 0D se coloca para indicar el final de una palabra. Una vez que lo encuentra, el procesador salta hacia adelante (a STOP), incrementando en uno al registro X y el número de palabra correspondiente en la tabla, quedando registrado en la dirección $FE. El mismo byte es suministrado por CHRGOT, saltando el procesador hacia atrás, al principio de la rutina ($C041). Así comienza la comparación con la siguiente palabra de la tabla.

Si no encontrase un $0D, se compara el byte de la tabla con 1, que indica el final de la misma. Una vez hallado el 1, el procesador bifurca hacia la rutina de salida, en la dirección $CF00.

Si la letra situada en el acumulador (A) no coincide con una de la tabla y la tabla no ha alcanzado el final de una palabra, resulta incrementado el registro X (que actúa como contador en este caso) saltando el procesador para comprobar de nuevo.

Una vez localizada la primera letra de un comando gráfico, debe ser comprobado el resto de letras de la nueva *palabra clave,* para ver si coincide con la de la tabla. En caso contrario se obtiene un mensaje de error de sintaxis.

```
*=$c061;ORG49249
ENC     INX
        LDA #$0D
        CMP $CF00,X
        BEQ $C089
        LDA #$01
        CMP $CF00,X
        BEQ $C089
        JSR $0073
        CMP $CF00,X
        BEQ ENC
        LDY $FB
        STY $7A
        LDY $FC
        STY $7B
        LDA $FE
        CMP #$16
        BEQ $C0A4
        JMP $C046
```

```
:C061 E8 A9 0D DD 00 CF F0 20
:C069 A9 01 DD 00 CF F0 19 20
:C071 73 00 DD 00 CF F0 E9 A4
:C079 FB 84 7A A4 FC 84 7B A5
:C081 FE C9 16 F0 1E 4C 46 C0
```

```
10 T=0
20 FOR I=49249 TO 49288
30 READ A
40 POKE I,A
50 T=T+A
60 NEXT I
70 IF T<>5615 THEN PRINT
   "ERROR EN LAS DATA"
100 DATA 232,169,13,221,0,
    207,240,32,169,1
110 DATA 221,0,207,240,25,
    32,115,0,221,0
120 DATA 207,240,233,164,
    251,132,122,164,252,132
130 DATA 123,165,254,201,22,
    240,30,76,70,192
```

Lo primero que se comprueba es la existencia de un $0D o un 1 en el siguiente byte de la tabla. Naturalmente que no se encontrará a ninguno en la primera pasada, pero estas instrucciones forman parte de un bucle. Cuando se encuentra cualquiera de ellos, se termina la comprobación y el procesador salta a la rutina que calcula que otra rutina es la necesaria para continuar.

El siguiente byte del programa escrito en BASIC se obtiene nuevamente partiendo de la rutina CHRGET, siendo comparado con el byte de la tabla. Si coinciden, el procesador vuelve al comienzo de esta rutina, comprobando el siguiente byte.

En caso negativo, el puntero del byte del BASIC es cargado en la variable del sistema adecuado. El número de palabra depositado en la dirección $FE es comparado con 22 ($16), para ver si han sido comprobadas todas las palabras. Si lo han sido, con BEQ $C0A4 se salta hacia la rutina de salida. De otra forma se vuelve a $C046, que está marcada con la etiqueta OTRVEZ en la rutina, para comenzar *otra vez* las comprobaciones.

¿QUE RUTINA?

Ahora el programa salta a la rutina correcta, para desarrollar el comando gráfico.

```
*=$C089;ORG49289
        LDX #$01
        LDY #$00
SUMA    CPX $FE
        BEQ $C097
        INX
        INY
        INY
        JMP SUMA
        LDA $C0C0,Y
        STA $FD
```

```
LDA $C0C1,Y
STA $FE
JMP ($00FD)
LDX #$0B
JMP ($0300)
```

```
:C089 A2 01 A0 00 E4 FE F0 06
:C091 E8 C8 C8 4C 8D C0 B9 C0
:C099 C0 85 FD B9 C1 C0 85 FE
:C0A1 6C FD 00 A2 0B 6C 00 03
```

```
10 T=0
20 FOR I=49289 TO 49320
30 READ A
40 POKE I,A
50 T=T+A
60 NEXT I
70 IF T<>4649 THEN PRINT
   "ERROR EN LAS DATA"
100 DATA 162,1,160,0,228,
    254,240,6,232,200
110 DATA 200,76,141,192,
    185,192,192,133,253,185
120 DATA 193, 192,133,254,
    108,253,0,162,11,108
130 DATA 0,3
```

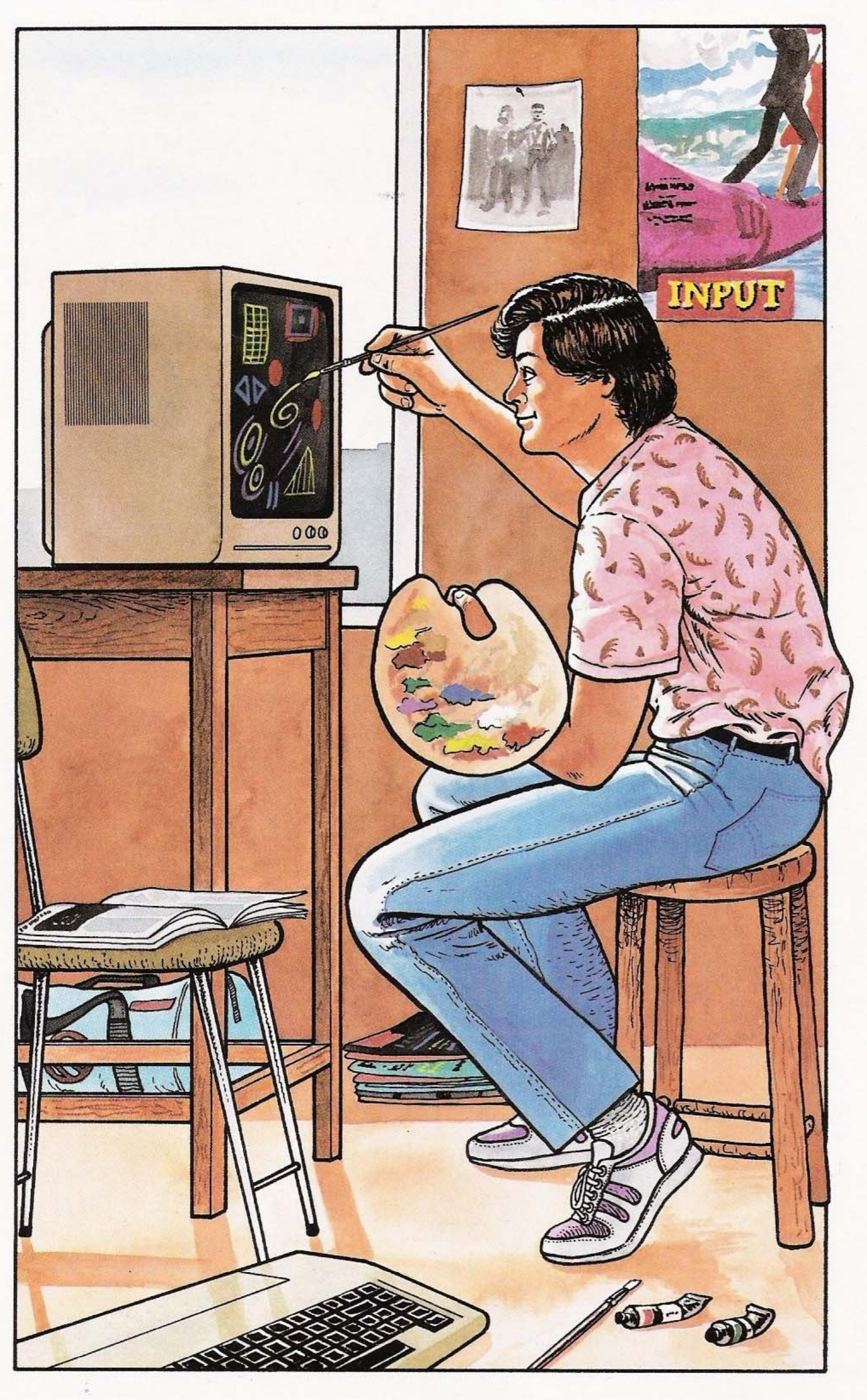
Los registros X e Y son inicializados e incrementados, hasta que el número de palabra depositado en $FE coincida con el valor de X. Entonces el valor calculado en Y se utiliza como «ajuste» para leer por toda la tabla que empieza en $C0C0, calculando la dirección de la rutina que se encargará de esa instrucción. La dirección inicial de la rutina está almacenada en las direcciones $FD y $FE. JMP ($00FD) salta hacia la rutina que acabamos de comentar.

LDX#$0B y JMP ($0300) hacen la llamada a una pequeña rutina de salida cuando la palabra del programa no se ajusta a ninguna de las que aparecen en la tabla. La rutina a la que apunta $0300 u $0301 devuelve el mensaje de error del BASIC, especificado por el número que se encuentra en el registro X.

LA TABLA DE RUTINAS

Aquí van las direcciones de inicio de las rutinas.

```
*=$C0C0;ORG49344
        .WOR $C100
        .WOR $C130
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
```



```
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
        .WOR $CE00
```

```
:C0C0 00 C1 30 C1 00 CE 00 CE
:C0C8 00 CE 00 CE 00 CE 00 CE
:C0D0 00 CE 00 CE 00 CE 00 CE
:C0D8 00 C8 00 C8 00 C8 00 C8
:C0E0 00 CE 00 CE 00 CE 00 CE
:C0E8 00 CE 00 CE
```

```
10 T=0
20 FOR I=49344 TO 49387
30 READ A
40 POKE I,A
50 T=T+A
60 NEXT I
70 IF T<>4554 THEN PRINT
   "ERROR EN LAS DATA"
100 DATA 0,193,48,193,0,
    206,0,206,0,206
110 DATA 0,206,0,206,0,206,
    0,206,0,206
120 DATA 0,206,0,206,0,206,
    0,206,0,206
130 DATA 0,206,0,206,0,206,
    0,206,0,206
140 DATA 0,206,0,206
```

En el presente artículo solamente abordaremos dos de los comandos gráficos. El resto serán tratados en siguientes capítulos. Por eso únicamente aparecen dos direcciones de comienzo de rutinas activas en la tabla-$C130 y $C100. Por el momento, si son encontradas otras palabras, serán dirigidas a una rutina temporal ubicada en $CE00. La dirección de comienzo de las otras rutinas gráficas serán escritas cuando tratemos con cada una de ellas.

LA RUTINA @ COLOUR

Esta parte del programa realiza varias llamadas a la ROM. JSR $B79B salta a la subrutina que atiende al primer parámetro que va tras el comando COLOUR. Este es transferido desde el registro X al acumulador y almacenado en la dirección de Entrada/Salida que controla el color de la pantalla.

```
*=$C100;ORG49408
        JSR $B79B
        TXA
        STA $D021
        JSR $AEFD
        JSR $B79E
        TXA
        STA $D020
        LDX $C00E
        JMP $0079
```

```
:C100 20 9B B7 8A 8D 21 D0 20
:C108 FD AE 20 9E B7 8A 8D 20
:C110 D0 AE 0E C0 4C 79 00
```

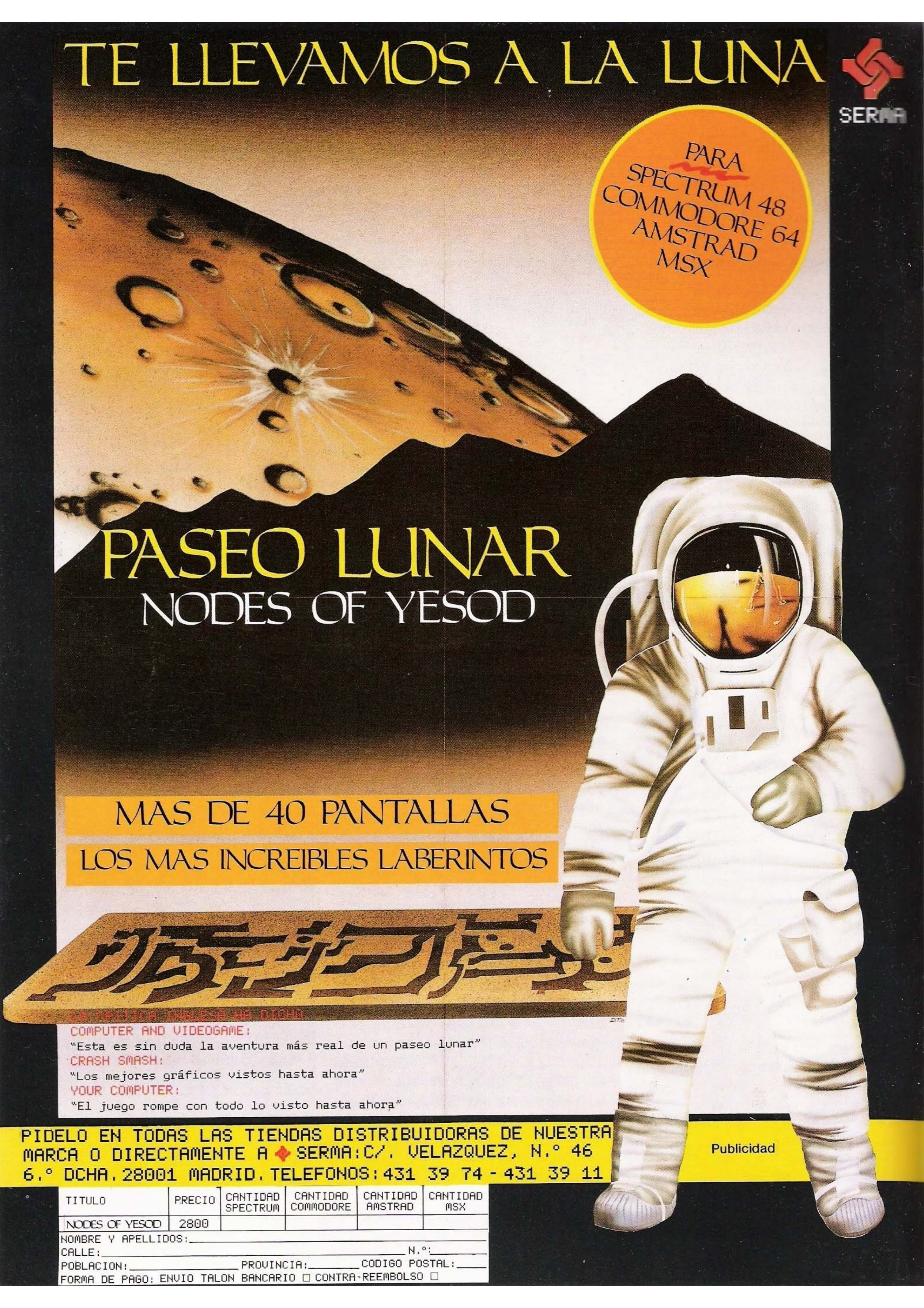
```
10 T=0
20 FOR I=49408 TO 49430
30 READ A
40 POKE I,A
50 T=T+A
60 NEXT I
70 IF T<>2818 THEN PRINT
   "ERROR EN LAS DATA"
100 DATA 32,155,183,138,
    141,33,208,32,253,174
110 DATA 32,158,183,138,
    141,32,208,174,14,192
120 DATA 76,121,0
```

La rutina de la ROM que comienza en $AEFD trata con las comas. En este caso se ignora la coma y es leído el siguiente parámetro, llamado en $B79E. Los tres primeros bytes de esa rutina lo desplazan hasta el siguiente byte, pero aquí no es necesario, ya que la rutina de la coma ya lo ha hecho. El byte es almacenado en la dirección que controla el color del borde. Entonces es cargado el estado de error en X y el procesador sale del programa por vía de la rutina de mensaje de error.

LA RUTINA @ HIRES

El registro de control del *chip* VIC es puesto en modo «mapa de bits» (o @ HIRES), mediante las instrucciones LDA #$3B y STA $D011. Las cuatro instrucciones siguientes preparan el registro de control de memoria del VIC, cambiándolo de 21 a 24, para trabajar en alta resolución.

```
*=$C130;ORG49456
        LDA #$20
        STA $FE
        LDA #$00
        STA $FD
BUC1    LDY #$00
BUCLE   STA ($FD),Y
        INY
        CPY #$00
        BNE BUCLE
        INC $FE
        LDX $FE
        CPX #$40
        BNE BUC1
        LDA #$3B
        STA $D011
        LDA $D018
        ORA #$08
        STA $D018
        JSR $B79B
        TXA
        AND #$0F
        ASL A
        ASL A
        ASL A
        ASL A
        STA $02
        JSR $AEFD
        JSR $B79E
        TXA
        AND #$0F
        CLC
        ADC $02
        STA $02
        LDA $02
        LDY #$00
REPITE  STA $0400,Y
        STA $04FA,Y
        STA $05F4,Y
        STA $06EE,Y
        INY
        CPY #$FB
        BNE REPITE
        LDA #$60
        STA $C23E
        JSR $C208
        LDA #$4C
        STA $C23E
```



TITULO	PRECIO	CANTIDAD SPECTRUM	CANTIDAD COMMODORE	CANTIDAD AMSTRAD	CANTIDAD MSX
NODES OF YESOD	2800				

NOMBRE Y APELLIDOS: ____
CALLE: ____ N.°: ____
POBLACION: ____ PROVINCIA: ____ CODIGO POSTAL: ____
FORMA DE PAGO: ENVIO TALON BANCARIO ☐ CONTRA-REEMBOLSO ☐

```
        LDA #$C8
        STA $D016
        LDX $C00E
        JMP $0079
```

```
:C130 A9 20 85 FE A9 00 85 FD
:C138 A0 00 91 FD C8 C0 00 D0
:C140 F9 E6 FE A6 FE E0 40 D0
:C148 EF A9 3B 8D 11 D0 AD 18
:C150 D0 09 08 8D 18 D0 20 9B
:C158 B7 8A 29 0F 0A 0A 0A 0A
:C160 85 02 20 FD AE 20 9E B7
:C168 8A 29 0F 18 65 02 85 02
:C170 A5 02 A0 00 99 00 04 99
:C178 FA 04 99 F4 05 99 EE 06
:C180 C8 C0 FB D0 EF A9 60 8D
:C188 3E C2 20 08 C2 A9 4C 8D
:C190 3E C2 A9 C8 8D 16 D0 AE
:C198 0E C0 4C 79 00
```

```
 10 T=0
 20 FOR I=49456 TO 49564
 30 READ A
 40 POKE I,A
 50 T=T+A
 60 NEXT I
 70 IF T<>13240 THEN PRINT
    "ERROR EN LAS DATA"
100 DATA 169,32,133,254,
    169,0,133,253,160,0
110 DATA 145,253,200,192,0,
    208,249,230,254,166
120 DATA 254,224,64,208,
    239,169,59,141,17,208
130 DATA 173,24,208,9,8,
    141,24,208,32,155
140 DATA 183,138,41,15,10,
    10,10,10,133,2
150 DATA 32,253,174,32,158,
    183,138,41,15,24
160 DATA 101,2,133,2,165,2,
    160,0,153,0
170 DATA 4,153,250,4,153,
    244,5,153,238,6
180 DATA 200,192,251,208,
    239,169,96,141,62,194
190 DATA 32,8,194,169,76,
    141,62,194,169,200
200 DATA 141,22,208,174,14,
    192,76,121,0
```

Después es introducido el siguiente parámetro, pasado al acumulador (A) y almacenado en $0286-la variable del sistema que contiene el código de color de fondo en cada instante dado. Seguidamente es controlada la presencia de una coma, saltando sobre ella para continuar leyendo el siguiente parámetro.

A continuación se carga la condición del código de error en el registro X y el procesador abandona el programa por vía de la rutina de visualización de error.

RUTINA DE NUEVA PALABRA

Esta rutina devuelve el mensaje de error.

La rutina que comienza en la dirección $FFD2 se encarga de visualizar en pantalla el carácter cuyo número se encuentra en el acumulador.

```
*=$CE00;ORG52736
        LDX #$00
        LDA $CE13,X
        JSR $FFD2
        INX
        CPX #$12
        BNE $CE02
        LDX $C00E
        JMP $0079
        .BYTE 'NO...IMPLEMEN
        TADO'
```

```
:CE00 A2 00 BD 13 CE 20 D2 FF
:CE08 E8 E0 12 D0 F5 AE 0E C0
:CE10 4C 79 00 4E 4F 2E 2E 2E
:CE18 49 4D 50 4C 45 4D 45 4E
:CE20 54 41 44 4F
```

```
 10 T=0
 20 FOR I=52736 TO 52771
 30 READ A
 40 POKE I,A
 50 T=T+A
 60 NEXT I
 70 IF T<>3767 THEN PRINT
    "ERROR EN LAS DATA"
100 DATA 162,0,189,19,206,
    32,210,255,232,224
110 DATA 18,208,245,174,14,
    192,76,121,0,78
120 DATA 79,46,46,46,73,77,
    80,76,69,77
130 DATA 69,78,84,65,68,79
```

LA TABLA EN ASCII

Esta tabla alberga a todas las *palabras clave* enforma de datos correspondientes al código ASCII. Cada palabra termina con un byte $0D y el final de la tabla lo hace con $01.

```
*=$CF00;ORG52992
        .BYTE 'COLOUR'
        .BYTE $0D
        .BYTE 'HIRES'
        .BYTE $0D
        .BYTE 'MULT'
        .BYTE $0D
        .BYTE 'NRM'
        .BYTE $0D
        .BYTE 'LOWCOL'
        .BYTE $0D
        .BYTE 'HICOL'
        .BYTE $0D
        .BYTE 'PLOT'
        .BYTE $0D
        .BYTE 'LINE'
        .BYTE $0D
        .BYTE 'TEST'
        .BYTE $0D
        .BYTE 'CSET'
        .BYTE $0D
        .BYTE 'REC'
        .BYTE $0D
        .BYTE 'CHAR'
        .BYTE $0D
        .BYTE 'TEXT'
        .BYTE $0D
        .BYTE ARC'
        .BYTE $0D
        .BYTE 'ANGL'
        .BYTE $0D
        .BYTE 'CIRCLE'
        .BYTE $0D
        .BYTE 'DRAW'
        .BYTE $0D
        .BYTE 'ROT'
        .BYTE $0D
        .BYTE 'FLASH'
        .BYTE $0D
        .BYTE 'OFF'
        .BYTE $0D
        .BYTE $01
```

```
:CF00 43 4F 4C 4F 55 52 0D 48
:CF08 49 52 45 53 0D 4D 55 4C
:CF10 54 0D 4E 52 4D 0D 4C 4F
```

```
:CF18 57 43 4F 4C 0D 48 49 43
:CF20 4F 4C 0D 50 4C 4F 54 4C
:CF28 49 4E 45 0D 54 45 53 54
:CF30 0D 43 53 45 54 0D 52 45
:CF38 43 0D 43 48 41 52 0D 54
:CF40 45 58 54 0D 41 52 43 0D
:CF48 41 4E 47 4C 0D 43 49 52
:CF50 43 4C 45 0D 44 52 41 57
:CF58 0D 52 4F 54 0D 46 4C 41
:CF60 53 48 0D 4F 46 46 0D 01
```

```
10 T=0
20 FOR I=52992 TO 53095
30 READ A
40 POKE I,A
50 T=T+A
60 NEXT I
70 IF T<>6621 THEN PRINT
   "ERROR EN LAS DATA"
100 DATA 67,79,76,79,85,82,
    13,72,73,82
110 DATA 69,83,13,77,85,76,
    84,13,78,82
120 DATA 77,13,76,79,87,67,
    79,76,13,72
130 DATA 73,67,79,76,13,80,
    76,79,84,76
140 DATA 73,78,69,13,84,69,
    83,84,13,67
150 DATA 83,69,84,13,82,69,
    67,13,67,72
160 DATA 65,82,13,84,69,88,
    84,13,65,82
170 DATA 67,13,65,78,71,76,
    13,67,73,82
180 DATA 67,76,69,13,68,82,
    65,87,13,82
190 DATA 79,84,13,70,76,65,
    83,72,13,79
200 DATA 70,70,13,1
```

COMPROBACION

Empleando este programa en BASIC con comandos gráficos, puedes comprobar que tu programa funciona:

```
10 @HIRES 0,1
20 FOR Z=8192 TO 16191 STEP
   5:POKE Z,255:NEXT Z
30 FOR Z=0 TO 15:@COLOUR Z,Z
   :NEXT Z: GOTO 30
```

Basta con teclear SYS 49152, para que se ejecute la rutina.

En los casos en que se manipule la pantalla utilizando los comandos gráficos de INPUT, los programas en BASIC más largos de 6 Kbytes no podrán ser ejecutados. Es por ello que los programas de tales características que publiquemos serán más cortos de esa longitud. De todas formas, si quisieras escribir sobrepasando este límite, tendrás que desplazar el área de memoria del BASIC al otro lado del área de la pantalla. Esto se consigue con los siguientes POKEs: POKE 43,1: POKE 44,65: POKE 54*256,0: NEW que proporciona 24 K de memoria libres para contener tu programa.

EL JUEGO DE LA VIDA

Cuéntale a tu microordenador unas cuantas reglas sencillas y podrás simular el comportamiento de los organismos unicelulares. El truco de este juego consiste en establecer una supervivencia óptima para la colonia.

En este juego la pantalla del televisor representa al mundo en el que pueden vivir, multiplicarse y morir las células. La pantalla queda dividida en cuadrados invisibles, cada uno de los cuales puede contener a una de las células. Si observamos el diagrama de la figura nos damos cuenta de que cada célula puede tener hasta ocho vecinos como máximo. Las reglas de la vida determinan si una célula vive o muere o, incluso, si nace una nueva.

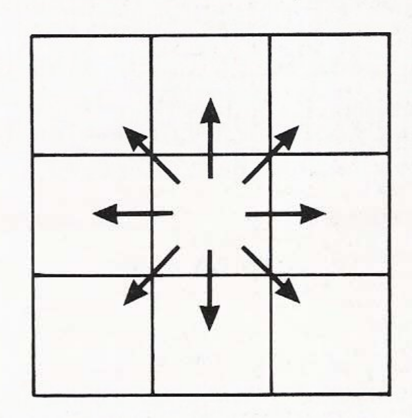
Las reglas son:

—Nacerá una célula en un espacio en el que existan exactamente tres vecinos.

—Una célula sobrevive hasta la siguiente generación si tiene dos o tres vecinos.

—El resto de las células muere.

El ordenador utiliza estas reglas para determinar el futuro de cada cuadrado de la pantalla y muestra cómo evoluciona una colonia inicial de células.



Cada célula tiene otras ocho como vecinas, de ellas depende su futuro.

Conseguir esto en BASIC sería terriblemente lento, por lo que el programa está desarrollado en código máquina, permitiendo que cada nueva generación pueda ser visualizada en aproximadamente un segundo.

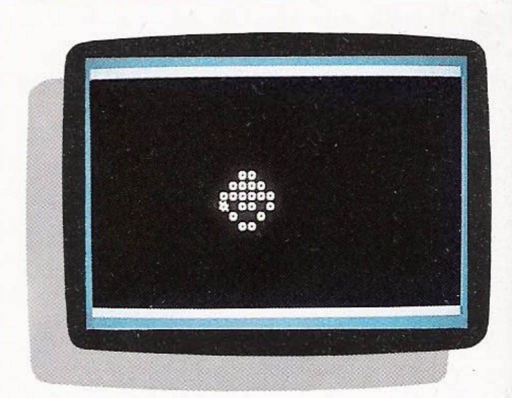
El programa también muestra en la pantalla a las sucesivas generaciones en un color diferente, por lo cual el efecto resulta bastante mesmerizante cuando observamos cómo se extiende la colonia original por la superficie, dividiéndose en grupos más pequeños, muriendo o rejuveneciéndose.

La conformación de la colonia inicial es de importancia crucial, pues resulta que algunos patrones conducen a la muerte tras la existencia de sólo unas pocas generaciones, mientras que otros perduran a lo largo de centenares. Tu trabajo consiste en introducir las células que componen la colonia inicial cuyos individuos pueden reproducirse y morir.

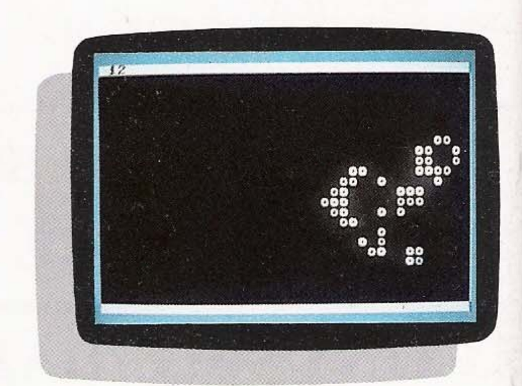
Uno de los objetivos de este juego es diseñar un patrón de partida que sobreviva durante el mayor número de generaciones. Con este fin existe un contador de generaciones que te mantiene informado sobre lo bien que lo está haciendo tu colonia. Pero tal vez puede ser más interesante crear un colonia que se desarrolla de manera intrigante. Las fotos de pantallas que acompañan al artículo muestran unos cuantos patrones obtenidos.

CREACION DE UNA COLONIA

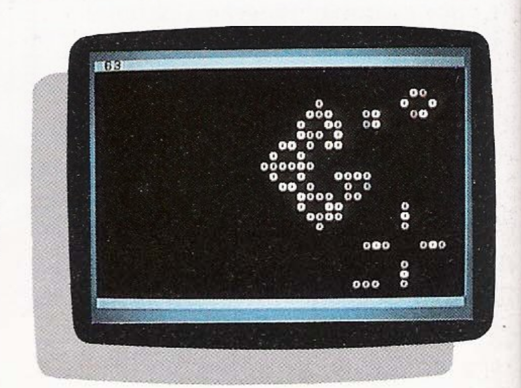
A continuación se expone cómo introducir el patrón inicial, es decir la colonia original. Para esto se utilizan las teclas de movimiento del cursor para desplazarse por toda la pantalla. La barra espaciadora para crear una célula, DEL para borrarla y RETURN para terminar.



El programador define la colonia o conjunto de ellas a su criterio, con el fin de que permanezcan con vida el mayor número de generaciones.



Al cabo de varias generaciones ya es factible estimar si la colonia tiene posibilidades de perdurar. En el caso de varias podrían fundirse.



La suerte está echada. Han nacido muchas células, otras son viejas y muchas se juegan el futuro a cara o cruz.

PERSONAL PRINTER

RITEMAN C+

BUILT-IN COMMODORE INTERFACE

INFORUNNER

RITEMAN C+

ESPECIFICACIONES

1. **Método de impresión**
   Serie de impactos por matriz de 9×9
2. **Cabezal impresor**
   9 agujas (reemplazable) con expectativa de vida de 50 millones de caracteres a 100 % de uso contínuo.
3. **Velocidad de impresión**
   Impresión a 105 cps uni o bi-direccional, búsqueda lógica optimizada, capacidad de impresión 45 lpm a 10 cpi., 200 ms de salto de línea con espaciado de 1/6".
4. **Espaciado interlínea**
   1/6", 1/8", 7/72", programable N/72", N/216".
5. **Alimentación de papel**
   a) Tractor ajustable entre 4" y 10".
   b) Alimentación a fricción para hojas sueltas entre 4" y 9,5".
6. **Corte de papel**
   Se puede imprimir desde principio de página hasta una pulgada de fin de papel.
7. **Selección de caracteres**
   96 caracteres ASCII, 96 itálicos, 82 caracteres gráficos Commodore. Definición de caracteres por usuario a través de software fácilmente.
8. **Gráficos**
   Normal 480 puntos por línea. Doble densidad 960.
9. **Modos de impresión**
   Modo Commodore
   Standard, reverse (negativo), mayúsculas y minúsculas y caracteres gráficos.
   Modo plus
   Standard, doble impresión, enfatizado, itálica, supra y subíndices mitad altura.
10. **Tamaños de impresión**
    Modo Commodore
    a) Normal
    b) Expandido
    c) Expandido doble
    d) Expandido triple
    e) Expandido cuádruple.
    Modo plus
    a) Normal (10 cpi), 2,1 mm (a)×2,55 mm(h).
    b) Normal expandido (5 cpi)
    c) Comprimido (17 cpi)
    d) Comprimido-expandido (8,5 cpi).
11. **Test escritura**
    Manteniendo apretado LF (salto de línea) cuando se acciona el interruptor de puesta en marcha ON.
12. **Subrayado contínuo.**
13. **Tabulador**
    Existen 32 posiciones de tabulación por línea.
14. **Longitud de formato**
    El standard de 66 líneas, con espaciado de líneas y longitud del formato programable.
    VFU eléctrico.
15. **Tamaño buffer**
    1 línea standard.
16. **Diagnóstico de detección de errores en modo Commodore.**
17. **Posicionador de papel contínuo**
    Desplazable parte anterior o posterior.
18. **Interface**
    Directo a Commodore salida RS232 a través de conector DIN (el cable se suministra con la impresora).
19. **Grosor papel**
    Máximo 2 copias, 40 g/m^2 peso contínuo, sensor papel.
20. **Operación con interruptores**
    De fácil acceso.
21. **Caracteres por línea**
    Normal: 80
    Normal expandido: 40
    Comprimido: 132
    Comprimido expandido: 66
22. **Cinta impresora**
    Autorretintada, mediante tampón interno, cartucho compacto de cinta negra fácilmente reemplazable y bajo coste.
23. **Tamaño**
    115 (h) × 275 (f) × 405 (a).
24. **Tensión alimentación**
    220 v/ 50 Hz.
25. **Entrada de papel**
    Frontal-horizontal.
26. **Retroceso**
    Por software.
27. **Instrucciones de formateado de tablas alfanuméricas**
28. **Elevadores inferiores standard**
    (para ubicación del papel contínuo debajo de la impresora).
29. **Motor: paso a paso**
30. **Indicadores:**
    - Conexión general
    - Fin de papel
    - On line
31. **Interruptores**
    ON/OFF, On line (SEL), salto de página (FF), salto de línea (LF).
32. **Compatible con la mayor parte de software existente** *(incluso Commodore).*

DATAMON
DATAMON, S. A.

REPRESENTACION EN ESPAÑA DE:



-IMPRESORAS PROFESIONALES-

- PERIFERICOS ORDENADORES
- IMPRESORAS Y MONITORES
- CONEXIONES Y ACCESORIOS

PROVENZA, 385-387, 6.°, 1.ª
TELÉFONO (93) 207 27 04 · 08025 BARCELONA

TODAS LAS ESPECIFICACIONES PUEDEN SER CAMBIADAS SIN PREVIO AVISO. * Commodore es marca registrada de Commodore Bussines Machine, Inc.

Teclea

```
  5 FOR Z=0 TO 278:READ
    X:POKE
    49152+Z,X:CO=CO+1:
    NEXT Z
  6 IF CO<>279 THEN
    PRINT"ERROR EN LAS
    DATA":STOP
 10 POKE 53280,6:POKE
    53281,0:POKE 198,0:POKE
    2,1
 20 PRINT"♡■":FOR Z=0 TO
    39:POKE 1024+Z,160:POKE
    1984+Z,160
 25 POKE 55296+Z,14:POKE
    56256+Z,14:NEXT Z
 30 X=500:C=32
 35 N=0:FOR Z=1064 TO
    1064+22*40 STEP 40
 40 POKE
    832+N,Z-(INT(Z/256)*256)
    :POKE 857+N,Z/256
 45 N=N+1:NEXT Z
100 GET A$
105 POKE 1024+X,C:IF
    A$=CHR$(13) THEN POKE
    198,0:GOTO 200
110 IF A$="▌" AND X-1>39
    THEN X=X-1
120 IF A$="◨" AND X+1<960
    THEN X=X+1
130 IF A$="□" AND X-40>39
    THEN X=X-40
140 IF A$="◪" AND X+40<960
    THEN X=X+40
150 C=PEEK(1024+X):POKE
    1024+X,102:POKE
    55296+X,3
160 IF A$="□" THEN C=87
170 IF A$=CHR$(20) THEN C=32
180 GOTO 100
190 PRINT"■■";NU:NU=NU+1:SY
    S 49152:IF PEEK(197)<>64
    THEN RUN 10
200 GOTO 190
210 DATA 169,0,133,251,32,
    159,192,166,251,189,
    64,3,133,252,133,254,
    189
220 DATA 89,3,133,253,24,
    105,190,133,255,160,0,
    177,252,201,32,208
230 DATA 16,200,192,40,208,
    245,230,251,165,251,
    201,23,208,216,76,178,
    192
240 DATA 224,0,240,12,192,
    0,240,8,192,39,240,4,
    224,23,208,3,76,34,192
250 DATA 136,177,254,24,
    105,1,145,254,200,200
    177,254,24,105,1,145,
    254
260 DATA 136,136,166,251,
    202,189,64,3,133,
    252,189,89,3,24,105,
    190,133
270 DATA 253,32,15,193,
    200,32,15,193,200,
    32,15,193,232,232,
    189,64,3,133
280 DATA 252,189,89,3,24,
    105,190,133,253,136,
    136,32,15,193,200,
    32,15
290 DATA 193,200,32,15,193,
    136,166,251,189,64,3,
    133,252,189,89,3,133,
    253
300 DATA 76,34,192,162,0,
    138,157,40,194,157,0,
    195,157,0,196,157,0,
    197,232
310 DATA 208,241,96,162,
    0,189,64,3,133,252,
    133,254,189,89,3,133,
    253,24
320 DATA 105,190,133,255,
    160,0,177,254,201,2,
    144,21,201,2,240,39,
    201,4
325 DATA 176,13,177,252
330 DATA 201,87,240,29,
    169,87,145,252,76,
    230,192,169,32,145,
    252,165,253
340 DATA 24,105,212,133,
    253,165,2,145,252,
    165,253,24,233,211,
    133,253
350 DATA 200,192,40,208,
    202,232,224,23,208,
    178,230,2,165,2,201,
    16
360 DATA 208,4,169,1,133,2,
    96,177,252,24,105,1,
    145,252,96
```

GANADORES DE LOS MEJORES DE INPUT

En el sorteo correspondiente al número 1 realizado entre quienes escribisteis mandando vuestros votos a LOS MEJORES DE INPUT han resultado ganadores:

Joan Campderros Canas (Barcelona)
Guillén Pera Blanco (Premia de Mar)
David Barcena Hilario (Madrid)
Xavier Parra Bernadet (Barcelona)
Jorge Brenuy Casals (Lérida)
Eduardo Quintana Coito (Barcelona)
María Crespo Lora (Nazaret)
Antonio López Soto (Málaga)
Xavier VI Vila y Capdevila (Solsona)
Federico Murcia Descalzo (Sevilla)

La Magia de...

Koala Pad™ Tableta Gráfica.

La Tableta Gráfica **KOALA PAD** pone a tu alcance las inmensas posibilidades gráficas del ordenador Commodore 64. Te abre las puertas a un mundo de creatividad y diversión donde no hay más límite que el de tu propia imaginación. Diseña y dibuja a mano alzada o utilizando la biblioteca de figuras geométricas. Con una paleta de 16 colores sólidos que, mediante variaciones de trama, proporcionan hasta 128 combinaciones posibles. Con 8 "pinceles" distintos, zoom (efecto lupa) para retocar detalles, efecto espejo... Conserva tus dibujos en disco o cassette. Imprímelos o inclúyelos en tus programas.

Con **KOALA PAD,** elegido "Periférico del Año" en Estados Unidos, se suministra el programa **KOALA PAINTER** en disco y cassette y el Manual de Usuario, ambos totalmente traducidos al castellano.

Y además, como Obsequio Especial, los programas:

KOALA PRINTER (cassette) para sacar por impresora las imágenes creadas por el **KOALA PAD.**

GUIA INSTANTANEA PARA EL PROGRAMADOR (cassette), una utilísima colección de herramientas de programación para que en tus programas puedas incluir la utilizacion del **KOALA PAD:** creación de cursores en alta resolución, zonas "sensibles" en pantalla, menús, teclas programables, "sprites", generación de tonos musicales...

P.V.P. 14.900ptas.







P.º de la Castellana, 179. 28046 MADRID. Teléfono: 442 54 33.

CONFIGURA LA HOJA DE TRABAJO (II)

- PLANEANDO LA HOJA DE TRABAJO
- USOS GENERALES
- FACTURAS
- INVENTARIOS

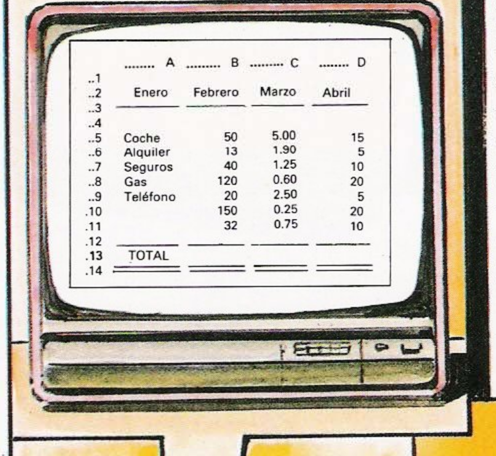
Mira adonde va a parar tu dinero, o planifica el futuro de tu negocio con este útil programa de hoja de trabajo. Agrega unas cuantas líneas al programa iniciado el mes pasado.

Ya que una hoja de trabajo al principio es como una hoja de papel en blanco (o una pantalla en blanco), a veces es difícil saber exactamente para qué utilizarla. Los ejemplos dados el mes pasado, y las ideas que siguen te ayudarán, de forma que podrás configurar tu hoja para tus propias necesidades. Y naturalmente no tienes por qué limitarte a una sola hoja. El programa se puede utilizar para crear cualquier número de hojas que pueden ser almacenadas y llamadas en cualquier momento.

Aquí tienes algunos ejemplos. Podrías tener una hoja para apuntar y planificar tus gastos domésticos, cuyas entradas irían encabezadas por rótulos tales como pagos, hipotecas, seguros, reparaciones, etc. Pero si tienes que hácer muchas reparaciones, es posible que prefieras mantener una hoja separada para ellas. Podrías etiquetar los diferentes tipos de reparaciones, así ocmo la decoración y los muebles de las distintas habitaciones, mostrando cuánto se ha gastado cada mes o cada trimestre. El programa podría totalizar el gasto para cada tipo de reparación en toda la casa, o el gasto total por habitación.

Otra hoja podría contener todos los gastos generales de la familia, tales como comida y bebida, ropa, coches, vacaciones y diversiones. Podría listar todos estos artículos por semanas o por meses, o para diferentes miembros de la familia.

Dado que la hoja de trabajo está completamente en blanco al empezar, puedes utilizarla para cualquier información, la cual ha de ser introducida de una forma lógica. Por ejemplo, la hoja de trabajo para los socios de un club puede listar sus nombres, números y cuotas de suscripción, así como las asistencias semanales, empleando etiquetas en casi todas las entradas.



Una hoja de trabajo resulta también ideal para introducir detalles en un informe. De hecho una hoja de trabajo es como una versión mucho más sofisticada de *array* bidimensional, permitiéndote un control mucho mayor sobre la forma en que se hacen las entradas así como sobre la inclusión de ecuaciones, cuyos resultados se calculan y presentan inmediatamente.

Las aplicaciones comerciales son más evidentes y virtualmente ilimita-



das. Las hojas de trabajo pueden contener detalles de facturas, características de artículos junto a su coste, IVA, descuentos, etc. Pueden servir para elaborar los salarios del personal, listando los nombres de los trabajadores, horas trabajadas, sueldo, subsidios e impuestos. También se pueden utilizar para la gestión de *stocks,* para la contabilidad o para la confección de presupuestos, incluyendo la elaboración de pronósticos alternativos para el futuro pulsando una simple tecla.

INTRODUCCION DEL PROGRAMA

La sección de programa que sigue a continuación enlaza con la que presentamos en el número anterior. El resto aparecerá en un próximo artículo, que también contendrá instrucciones detalladas sobre la forma de usar el programa. Carga pues la primera parte (con LOAD), añádele estas líneas y almacénalas (SAVE) listas para la próxima vez en que ya podrás ejecutar el programa completo.

Teclea para Commodore-64

```
860 A$=A$+"[ESPACIO]"
870 A$=LEFT$(A$,LEN(A$)-1)
880 IF RIGHT$(A$,1)="[ESPA
    CIO]" THEN 870
890 PS=1:LN=LEN(A$):K=0
900 IF LN=1 AND A$="R" THEN
    F1=3:RETURN
910 IF LN=1 THEN F1=0:
    RETURN
920 IF LN=0 AND RIGHT$(D$(R,
    C),8)="[8*ESPACIOS]"THEN
    A$=CHR$(128):F1=3:RETURN
930 GOSUB 790
940 PS=PS+1+D1
950 IF D1=0 THEN F1=0:RETURN
960 IF PS<LN-1 THEN 930
970 IF PS>LN THEN F1=3:
    RETURN
980 Q1$=OP$:Q2$=MID$(A$,PS,
    1):GOSUB 2670
990 IF Q=0 THEN F1=0:RETURN
1000 D=VAL(RIGHT$(A$,1))
1010 IF LN=PS THEN F1=3:
     RETURN
1020 IF LN=PS+1 AND D>1 AND
     D<10 THEN F1=3:RETURN
1030 F1=0:RETURN
1040 IF A$="" THEN RETURN
1050 GOSUB 860
1060 IF F1=0 THEN A$="":
     INPUT"[CRSRabajo]ESCRI
     BE[CRSRderecha]";A$:
     GOTO 1040
1070 RETURN
1080 PRINT"Q"SPC(16)"[CTRL+9
     ]TRABAJANDO[CRSRabajo]"
1090 FOR C=1 TO CM
1100 FOR R=1 TO RM
1110 D=0:F=0:OP=8
```

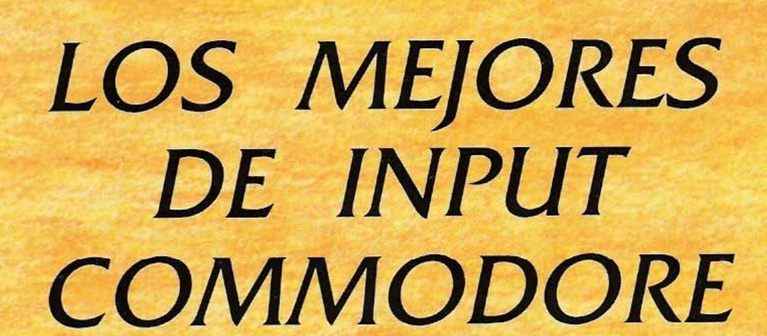


```
1120 F$=LEFT$(D$(R,C),8)
1130 C1=ASC(MID$(F$,1,1))-64
1140 IF C1=64 OR C1=-32
     THEN 1230
1150 IF C1=18 THEN GOSUB
     1320:GOTO 1230
1160 R1=VAL(MID$(F$,2,2)):IF
     R1>9 THEN F=1
1170 C2=ASC(MID$(F$,3+F))-64
     :IF C2=-32 THEN GOSUB
     1360:GOTO 1230
1180 IF C2=-3 THEN D=VAL
     (MID$(F$,4+F,2)):GOSUB
     1360:GOTO 1230
1190 R2=VAL(MID$(F$,4+F,2)):
     IF R2>9 THEN F=F+1
1200 Q1$=OP$:Q2$=MID$(F$,5+
     F,1):GOSUB 2670
1210 D=VAL(MID$(F$,6+F,2))
1220 GOSUB 1360
1230 NEXT R,C
1240 RETURN
1250 GET A$:IF A$="" THEN
     1250
1260 IF A$<AA$ OR A$>BB$
     THEN 1250
1270 RETURN
1280 GET A$:IF A$="" THEN
     1280
1290 IF A$=AA$ THEN RETURN
1300 IF A$=BB$ THEN RETURN
1310 GOTO 1280
1320 AA$=RIGHT$(D$(R,C),8)
1330 IF RIGHT$(AA$,1)="[ESPA
     CIO] " THEN AA$=LEFT$
     (F$,7):GOTO 1330
1340 D$(R,C)=LEFT$(D$(R,C),
     8)+AA$
1350 RETURN
1360 V1=VAL(RIGHT$(D$(R1,C1)
     ,8)):V2=VAL(RIGHT$(D$
     (R2,C2),8))
1370 ON Q GOSUB 1420,1430,
     1440,1450,1460,1470,
     1510,1550
1380 AA$=STR$(AN):IF LEN
     (AA$)>8 THEN AA$="DEMA
     SIADO GRANDE"
1390 IF LEN(AA$)<8 THEN AA$=
     "[ESPACIO] "+AA$:GOTO
     1390
1400 D$(R,C)=F$+AA$
1410 RETURN
1420 AN=V1+V2:RETURN
1430 AN=V1-V2:RETURN
1440 AN=V1*V2:RETURN
1450 AN=V1/V2:RETURN
1460 AN=V2*V1/100:RETURN
1470 AN=0:FOR NN=R1 TO R2
1480 AN=AN+VAL(MID$(D$(NN,
     C1),9,8))
1490 NEXT NN
1500 RETURN
1510 AN=0:FOR NN=C1 TO C2
1520 AN=AN+VAL(MID$(D$(R1,
     NN),9,8))
1530 NEXT NN
1540 RETURN
1550 AN=V1:RETURN
1560 PRINT"QUIERES ALMACENAR
     ESTOS DATOS (S/N)?"
1570 AA$="S":BB$="N":GOSUB
     1280
1580 IF A$="N" THEN RETURN
1590 INPUT"sNOMBRE";NM$:OPEN
     1,1,1,NM$
1600 PRINT#1,RM:PRINT#1,CM:
     FOR R=1 TO RM
1610 FOR C=1 TO CM
1620 PRINT#1,CHR$(34)+D$(R,
     C)+CHR$(34)
1630 NEXT C,R:CLOSE 1:RETURN
1640 PRINT"[SHIFT+CLR/HOME]
     [CRSRabajo][CRSRderecha]
     QUIERES CARGAR"
1650 PRINT"[CRSRderecha]UN
     FICHERO EXISTENTE
     (S/N)?"
1660 AA$="S":BB$="N":GOSUB
     1280
1670 IF A$="N" THEN RETURN
1680 INPUT"NOMBRE";NM$:OPEN
     1,1,0,NM$
1690 INPUT#1,RM,CM:FOR R=1
     TO RM:FOR C=1 TO CM
1700 INPUT#1,D$(R,C):NEXTC,R
1710 CLOSE 1:RETURN
1720 S$=CHR$(13)+CHR$(145)+"
     [22*ESPACIOS][22*SHIFT+
     CRSRizquierda]"
1730 W$="MAL MODO PARA
     RELATIVO "
1740 GOSUB 590
1750 IF A$="[FLECHAizq.]"
     THEN RETURN
1760 PRINT"[CRSRarriba]
     ABSOLUTO O RELATIVO";
1770 AA$="A":BB$="R":GOSUB
     1280
1780 IF TP=0 AND A$="R" THEN
     PRINT S$;W$;:FOR W=1 TO
     1000:NEXT
1790 IF TP=0 AND A$="R" THEN
     RETURN
1800 IF A$="A" THEN K1=0
1810 IF A$="R" THEN K1=1
1820 PRINT S$;
1830 PRINT"COPIA EN FILA/
     COLUMNA";
1840 AA$="C":BB$="F":GOSUB
     1280
1850 IF A$="C" THEN K2=1
1860 IF A$="F" THEN K2=0
1870 PRINT S$;
1880 PRINT"[SHIFT+CRSR
     arriba]DESDE COLUMNA
     [ESPACIO][CRSRderecha]";
1890 AA$="A":BB$="X":GOSUB
     1250
1900 C1=ASC(A$)-64:PRINT A$
1910 INPUT"FILA";R1
1920 IF R1<0 OR R1>CM THEN
     PRINT TAB(6)"[SHIFT+
     CRSRarriba";:GOTO
     1910
1930 PRINT"[SHIFT+CRSRarriba]
     HASTA[3*ESPACIOS]";
1940 IF K2=0 THEN AA$="A":
     BB$="W":GOSUB 1250:C2=
     ASC(A$)-64:PRINT A$;:
     R2=R1
1950 IF K2=1 THEN C2=C1:
     PRINT CHR$(C2+64):INPUT
     "FILA[ESPACIO];
     ";R2
1960 IF R2<R1 OR R2>RM THEN
     1950
1970 ON K1*2+K2+TP/2+1 GOSUB
     1990,2020,2070,2070,
     2080,2110,2140,2260
1980 RETURN
1990 GOSUB 2400
2000 FOR NN=C1 TO C2:GOSUB
     2450:NEXT NN
2010 RETURN
2020 GOSUB 2400
2030 FOR NN=R1 TO R2
2040 A$=LEFT$(D$(NN,C1),8):
     IF ASC(A$)=128 THEN A$=
     "[ESPACIO]"+RIGHT$(A$,
     7)
2050 D$(NN,C1)=A$+F$
2060 NEXT NN
2070 RETURN
2080 GOSUB 2410
```

```
2090 FOR NN=C1 TO C2:GOSUB 2470:NEXT NN
2100 RETURN
2110 GOSUB 2410
2120 FOR NN=R1 TO R2:GOSUB 2440:NEXT NN
2130 RETURN
2140 GOSUB 2410:GOSUB 2420
2150 FF=0:IF X1>9 THEN FF=1
2160 AA$=MID$(F$,2,1+FF)
2170 REM**CONTINUA EN EL SIGUIENTE
```

Nota: En algunos ejemplares de nuestra edicion anterior aparecieron cegados varios caracteres graficos, debido a un exceso de tinta. Por ello publicamos a continuacion la correspondencia de ellos, escribiendo entre corchetes las teclas o sus combinaciones necesarias en cada momento.
La operacion es sencilla. Si se hubiera tecleado la primera parte del programa. Basta con cargarlo desde la cinta o el disco y escribir directamente estas lineas, que pasaran automaticamente a sustituir a las anteriores.

```
20 PRINT "[SHIFT+CLR/HOME][CTRL+4][CRSR
   abajo]"SPC(16)"[CTRL+9]TRABAJANDO
   [CRSRabajo]"CHR$(8)
100 C$="[CRSRderecha][SHIFT+CRSRizq.]
    [CRSRabajo][SHIFT+CRSRarriba]"+CHR$
    (133)+CHR$(137)+CHR$(134)+CHR$(138)
    +CHR$(135)+CHR$(139)
220 PRINT"[CLR/HOME][CTRL+9][CTRL+5]
    [SHIFT+U][SHIFT+*][SHIFT+*]";
280 PRINT"[CTRL+9][SHIFT+B][COMM.+1]"
    RIGHT$(AA$,2);:N=0
300 PRINT"[CTRL+5][SHIFT+B][CTRL+0]"
    CHR$(CL(N));
350 PRINT"[CTRL+9][CTRL+5][SHIFT+B]";:
    NEXT R
360 POKE 198,0:PRINT"[CTRL+9][SHIFT+J]
    [2*SHIFT+C][COMM.+E][8*SHIFT+C]
    [COMM.+E][8*SHIFT+C][COMM.+E][8*
    SHIFT+C][COMM.+E][8*SHIFT+C][SHIFT+
    K]";
370 PRINT"[CTRL+9][ESPACIO]CURSOR PARA
    DESPLAZAR:";
390 IF TP=0 THEN PRINT"[ESPACIO][CTRL+8]
    [ESPACIO]MODO VARIABLES[ESPACIO]
    [CTRL+5][ESPACIO]";
640 PRINT"[SHIFT+CRSRizq.]"A$
690 PRINT"ENTRADA[CRSRder.]";
750 IF LEN(A$)>8 THEN A$=LEFT$(A$,8)
810 IF BB$<"A" OR BB$>"X" THEN D1=0:
    RETURN
```



LOS MEJORES DE INPUT COMMODORE

PUESTO	TITULO	PORCENTAJE
1.°	Summer Games II	19,67 %
2.°	Pist Top II	11,24 %
3.°	The Dambusters	11,04 %
4.°	Misión Imposible	10,04 %
5.°	Gi Joe	9,83 %
6.°	Sky Fox	9,23 %
7.°	Frankie goes to Hollywood	9,03 %
8.°	Le Tour de France	7,63 %
9.°	The Way of exploding fist	6,22 %
10.°	Panorama para matar	6,07 %
		100 %

Para la confección de esta relación únicamente se han tenido en cuenta las votaciones enviadas por nuestros lectores de acuerdo con la sección «Los Mejores de Input».

Noviembre de 1985



JUEGOS DE INVIERNO

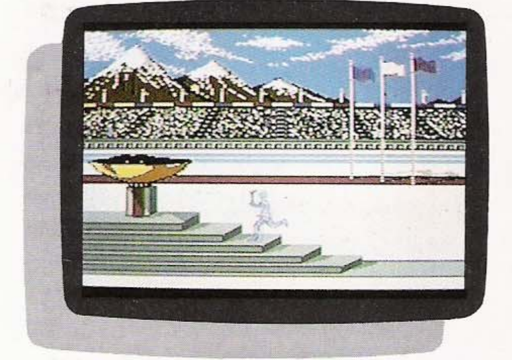
No es preciso esperar a la llegada de las primeras nieves para practicar los deportes de invierno. Este nuevo programa de **Epix** nos permite competir y practicar en una o varias de las distintas pruebas clásicas de las estaciones más frías.
Igual que en **Summer Games,** existe una ceremonia de apertura y tenemos acceso a los registros de los *records* mundiales de las especialidades contenidas en el programa.
Las opciones van eligiéndose por mediación de menús y el movimiento del *joystick*.
Podríamos comenzar con los saltos de *ski*. Es posible controlar el momento del salto desde la rampa de lanzamiento. Igualmente, el control de la palanca nos permite mantener el equilibrio, desplazando el centro de gravedad. Es muy fácil caer de bruces por inexperiencia. De cualquier modo, a la panorámica general se superpone una ventana por la que observar la posición del saltador con mayor nivel de detalle.

Una nueva rampa sirve para ejecutar esas difíciles y arriesgadas piruetas con *skies,* que se ven en las películas de acción con paisaje invernal. Es el *joystick* nuevamente el que nos ayuda a controlar el centro de gravedad del deportista.
Manejándolo con habilidad podemos conseguir piruetas realmente sorprendentes.
El patinaje sobre hielo ofrece tres modalidades. La primera es de figuras. Se pueden ejecutar siete movimientos diferentes, elegidos por la posición del *joystick*. Desde el *triple axel,* al *giro sentado,* sin olvidar el *doble salto lutz*. También es posible patinar de espalda.
La segunda modalidad es libre. Se pueden, combinar las figuras comentadas anteriormente, creando una coreografía propia. Pero, cuidado, aquí también es fácil caer.
Las dos primeras modalidades de ejecución son realizadas por una bella patinadora, mientras que la tercera, el patinaje de velocidad es acometida por dos rudos patinadores. Cada uno dispone de una mitad de la pantalla, ocupando el centro de cada pista el competidor en cuestión y el otro aparece o desaparece según la distancia que les separe. La velocidad se incrementa en la medida en que el ritmo seguido en el movimiento de la palanca se adecue al avance del patinador.
Para concluir quedan dos especialidades poco difundidas en nuestro país. Uno es el *ski* de fondo, en el que avanza el esquiador en solitario, ayudándose con los bastones (nuevamente el ritmo de movimiento del *joystick* juega un papel fundamental) por tres bonitos paisajes nevados.
Las carreras de *bobsled* ponen punto final al número de competiciones.











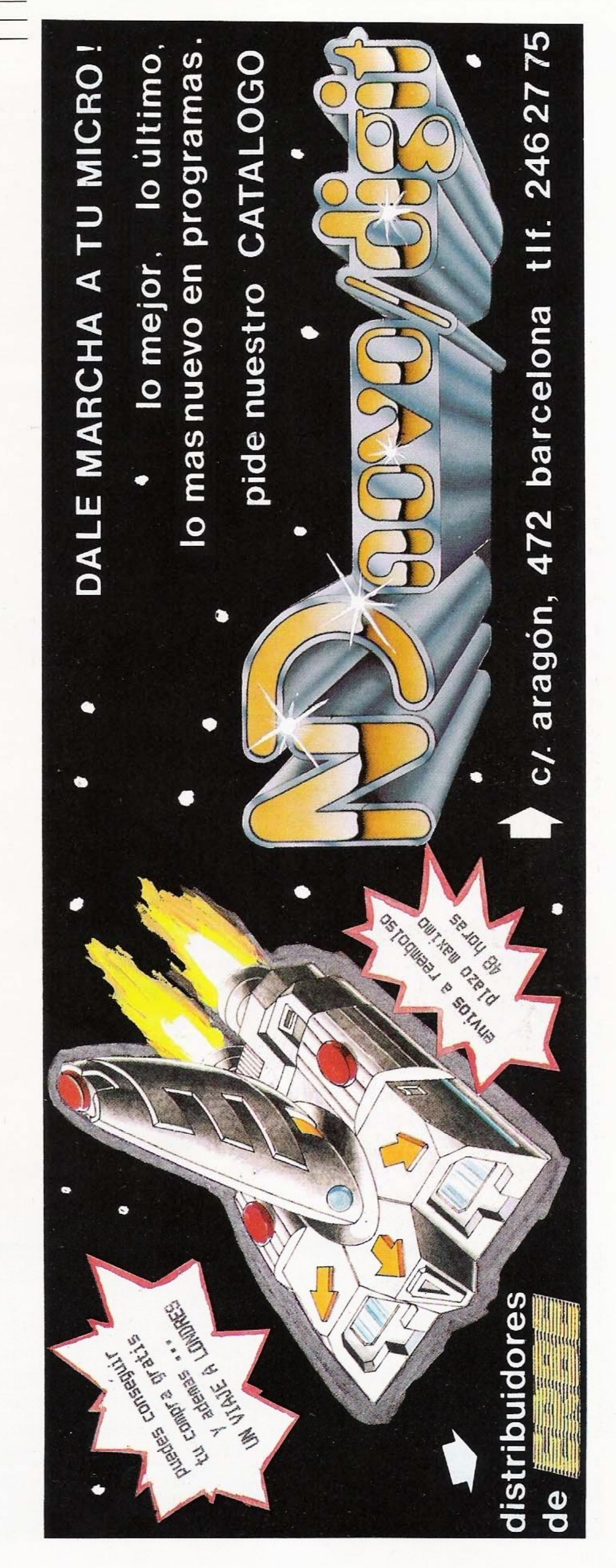
DATOS GENERALES	CALIFICACION
TITULO: Winter Games	GRAFICOS: 5 sobre 5
FABRICANTE: Epix	COLOR: 5 sobre 5
ORDENADOR: Commodore	PRESENTACION: 4,5 sobre 5
MEMORIA: 64	INTERES: 4 sobre 5
CLASE DE PROGRAMA: Competición	REALISMO: 4,5 sobre 5

Son esos coches tipo cajón que se deslizan por un semitubo de hielo y nieve, alcanzando velocidades de vértigo. Es preciso controlar hacia dónde se desplaza el peso en las curvas. Si no se hace correctamente, el vuelco está prácticamente asegurado. Nada se puede decir contra la calidad de los gráficos y el nivel de realismo de estas pruebas. **Winter Games** es un programa sencillamente excepcional.

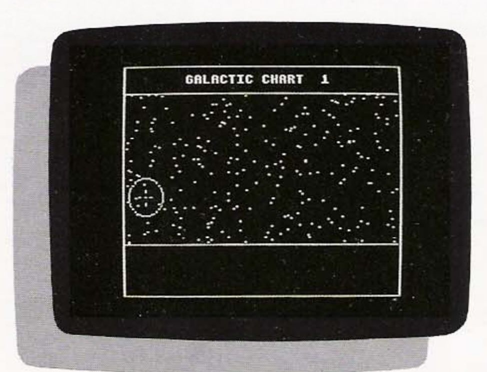
****

EL NAVEGANTE ESPACIAL

Eres el **Comandante Jameson** y estás al mando de la nave **Cobra MK III,** apta para el comercio y el combate.
En ella viajas por el espacio comprando, vendiendo y transportando toda clase de mercancías. Existe una tabla, que aparece en la pantalla, de los productos que puedes comprar, cantidades y precio unitario. Aquí tienes desde platino, maquinaria y textiles hasta armas de fuego o pieles.
En los dos mil planetas, repartidos entre ocho galaxias, hay de todo, incluso piratas no definidos como criminales, aunque sí como comerciantes fracasados que tratarán de arrebatarte la carga.
Es importante conocer el perfil de cada planeta antes de comerciar con él, en algunos reina de anarquía. Igualmente existen mundos agrícolas y mundos industriales. Es preciso tener todo esto en cuenta, antes de tomar una decisión. El agrícola demostrará gran avidez por las mercancías industriales, mientras que el industrial precisará comida.
La pantalla de mando es tremendamente completa. En condiciones normales consigues una vista frontal, aunque puedes conseguir las vistas izquierda, derecha y posterior para averiguar a qué, o quiénes se aproximan a la nave.
En la franja inferior de la pantalla se han dispuesto los paneles de los instrumentos de vigilancia y medida. En la izquierda vemos el termómetro que registra la temperatura interior de la cabina.
El altímetro entra en funcionamiento cuando orbitamos en torno a un planeta, y el velocímetro es de importancia crucial para las maniobras de acercamiento, entrada en órbita o el atraque para cargar o descargar mercancías.
El giróscopo nos indica nuestra posición relativa, si la nave pica o se inclina a la izquierda o derecha. El nivel de combustible y la brújula también están presentes.
La parte central está ocupada por el valioso *scanner*. Produce una visión tridimensional del entorno de la nave, pudiendo localizar y determinar la posición de cualquier nave que se nos aproxime.
Para orientarnos en el espacio, disponemos de una carta de navegación galáctica, que puede visualizarse para









ayudarnos a localizar los mundos favorables a nuestro comercio.
Las naves espaciales que irás encontrando son dibujadas en forma de líneas esquemáticas, que no le restan emoción al juego y, sin embargo, le producen una buena sensación tridimensional. Existen unas dieciocho de ellas, teniendo cada cual una fisonomía muy característica.
El juego no es fácil de dominar en las primeras etapas, pero es elevadamente absorvente para quien se enfrente a él. Es condición necesaria leerse el manual de instrucciones, en inglés, pero no de difícil comprensión dado su nivel de descripción detallada de todos los aspectos precisos, para dominar el comercio intergaláctico.
Una corta novela, nos proporciona un breve esbozo de como es la vida de estos camioneros espaciales.

DATOS GENERALES	CALIFICACION
TITULO: Elite	GRAFICOS: 4 sobre 5
FABRICANTE: Firebird	COLOR: 3 sobre 5
ORDENADOR: Commodore	PRESENTACION: 4,5 sobre 5
MEMORIA: 64	INTERES: 4 sobre 5
CLASE DE PROGRAMA: Aventura espacial	REALISMO: 4 sobre 5

****

PERDIDO EN EL GRAN ALMACEN

El pequeño **Herbert** se ha perdido. Se entretuvo mirando los juguetes y cuando se quiso dar cuenta sus padres ya no estaban a su lado. Ayúdale a encontrar a sus padres que le aguardan en el departamento de objetos perdidos con una bolsa repleta de golosinas.
La resolución gráfica, el colorido y el movimiento del personaje son tres de las características más destacadas de este excelente juego creado por **Mikro-Gen,** que además tiene todas sus indicaciones en castellano.
Herbert se encuentra solo en unos grandes almacenes que semejan un gran caserón con grandes salas que deberá atravesar hasta volver a encontrar a sus padres. Partiendo de la sala de juguetes el pequeño inicia su recorrido teniendo que evitar los





muchos peligros que en cada habitación le acechan. A medida que va avanzando surgen ante él una serie de objetos como llaves, linternas, transistores, cámaras fotográficas, etc. que debes seleccionar, ya que sólo puede llevar dos de estos objetos encima. Estos pueden ser muy útiles en determinados momentos, por lo que es conveniente elegirlos adecuadamente.
La pantalla muestra la sala en la que te encuentras y en la parte superior indica las vidas que te quedan, los objetos que lleva encima y el tiempo transcurrido. Para encontrar a sus padres **Herbert** deberá

atravesar el invernadero, la sala de música, el guardarropa, el dormitorio, la biblioteca, la sala de deportes, el salón principal y otras muchas habitaciones. Tu puntuación vendrá definidia por el número de golosinas que hayas recogido, así como el tiempo que hayas conseguido permanecer, añadiéndote un calificativo por no haber conseguido tu objetivo.





DATOS GENERALES	CALIFICACION
TITULO: Herbert's	GRAFICOS: 5 sobre 5
FABRICANTE: Mikro-Gen	COLOR: 5 sobre 5
ORDENADOR: Commodore	PRESENTACION: 4,5 sobre 5
MEMORIA: 64	INTERES: 4 sobre 5
CLASE DE PROGRAMA: Odisea infantil	REALISMO: 4 sobre 5

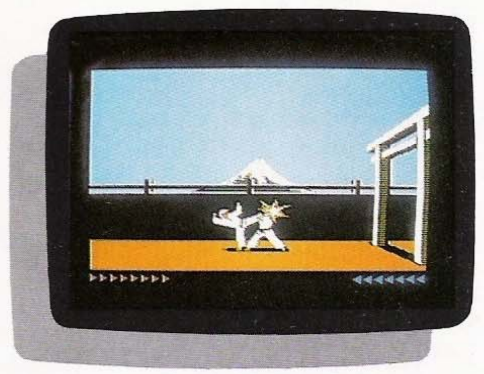
El consejo que te podemos dar es que evites el contacto con cualquier objeto que se mueva y aunque localices la salida permanece en el edificio, en la calle cae un fuerte chaparrón que puede acabar con tus energías.
Pórtate como un buen amigo y ayuda a **Herbert** a encontrar a sus padres, seguro que él te lo agradecerá y tú disfrutarás con unas magníficas pantallas.

****

KARATE

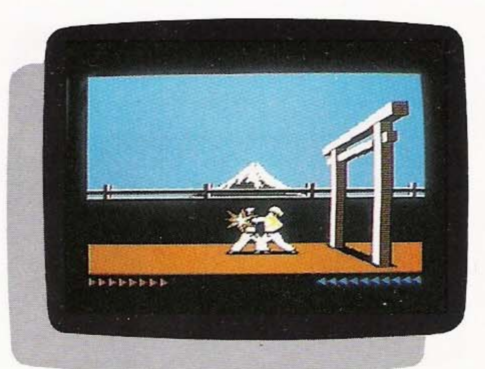
El tema principal de este juego tiene como fondo una tragedia oriental, donde el villano es el causante de todos los males y el bueno trata de «desfacer los entuertos».
El «chico» regresa contento a su casa tras haber pasado largos años aprendiendo y perfeccionándose en el antiguo y noble arte marcial que lleva por nombre karate. Su contento acaba pronto pues encuentra su aldea quemada y arrasada, su familia dispersada y, lo más importante, su prometida ha sido raptada por **Akuma,** el todo poderoso señor de turno de la zona.
Ahora tú tomas el protagonismo: debes escalar el acantilado y derrotar a los guardias, hasta enfrentarte mano a mano con **Akuma,** sólo así puedes rescatar a la princesa **Mariko.** Para ello no cuentas más que con tu valor, tus conocimientos de karate y... un *joystick*.
Tienes una seria limitación: no puedes dar más que un numero limitado de golpes, indicado en cada momento por una línea de triángulos que aparecen en la base de la pantalla, desapareciendo uno por golpe.

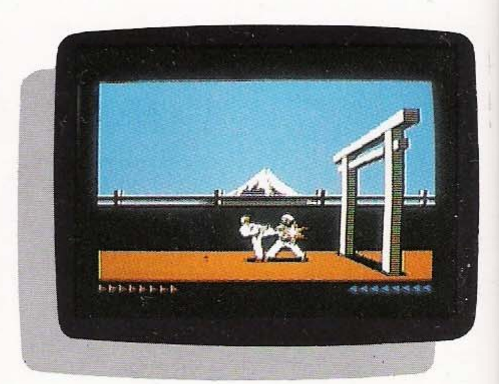




DATOS GENERALES	CALIFICACION
TITULO: Karateka	GRAFICOS: 5 sobre 5
FABRICANTE: Broderdbund	COLOR: 4,5 sobre 5
ORDENADOR: Commodore	PRESENTACION: 4,5 sobre 5
MEMORIA: 64	INTERES: 4 sobre 5
CLASE DE PROGRAMA: Artes marciales	REALISMO: 5 sobre 5







Puedes dar tanto patadas como puñetazos en tres áreas del enemigo; arriba, abajo y en el centro. Si calculas correctamente la distancia, cada golpe tendrá éxito, pero el enemigo también golpea y, por tanto tendrás que esquivar sus ataques. El control se puede realizar también por teclado, pero resulta notablemente más incómodo.

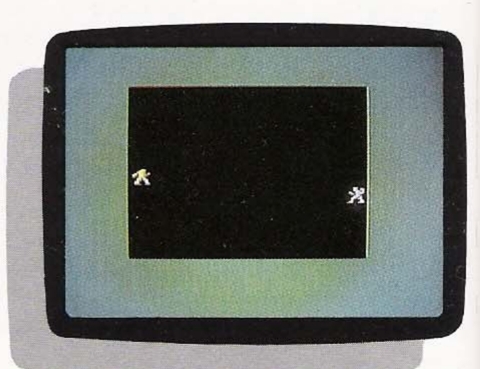
A juzgar por expertos en karate que han visto el desarrollo del juego, los movimientos son absolutamente realistas, de eso no cabe la menor duda.

****

EL AJEDREZ MITOLOGICO

Una de las diferencias más importantes que se aprecian con respecto al ajedrez es que si bien en éste una pieza come a la otra de un modo directo, en el **Archon** la cosa no es tan sencilla. Las piezas se desafían entre sí. Los jugadores eligen las piezas que se han de enfrentar y el tablero desaparece, siendo sustituido por un campo de batalla. Es aquí donde luchan con fiereza, utilizando todos sus recursos, los unicornios, valkirias, basíliscos, hechiceros, etc. La habilidad de los jugadores cuenta para neutralizar o destruir al adversario.
Los personajes tienen limitaciones. Algunos no pueden desplazarse más que un número determinado de casillas. Otros no pueden sobrevolar a otras piezas, por estar condenados a moverse sólo por tierra.
El color de las casillas se altera durante el transcurso del juego, beneficiando a la pieza que lo ocupa, proporcionándole mayores recursos y defensas para defenderse y atacar al enemigo.
El objetivo final consiste en ocupar cinco casillas estratégicas, llamadas «puntos de poder». Quien antes lo consiga, será el ganador. La otra manera de ganar consiste en deshacerte de todas las piezas del contrincante.





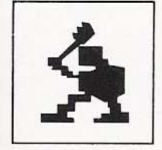
Este ajedrez es algo más que un juego, en el que las capacidades estratégicas de los participantes cuentan más de lo que pueda parecer. Igual que ocurre con el ajedrez normal, se pueden lograr distintos grados de maestría.
Cabe resaltar la enorme gracia con lo que evolucionan las figuras por el campo de batalla.



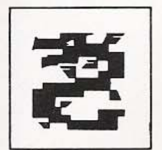
DATOS GENERALES	CALIFICACION
TITULO: Archon	GRAFICOS: 4,5 sobre 5
FABRICANTE: Ariola soft	COLOR: 3,5 sobre 5
ORDENADOR: Commodore	PRESENTACION: 5 sobre 5
MEMORIA: 64	INTERES: 5 sobre 5
CLASE DE PROGRAMA: Estrategia	REALISMO: 4 sobre 5

Una creación de Electronic Arts producida y presentada por DRO SOFT con instrucciones y pantallas en CASTELLANO.

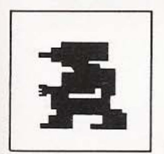
OGRO



DRAGON



TROLL



BASILISCO



BRUJA





FENIX



UNICORN



ARQUER



MAGO



CABALLE



En la época actual, estamos viviendo el resurgimiento del género de la fantasía y la aventura, dentro de esta corriente, JON FREEMAN, el principal creador de este juego elaboró un ajedrez fantástico cuyos personajes podían estar sacados de cualquier aventura de mazmorras y dragones (DUNGEONS AND DRAGONS).

ARCHON.

ARCHON presenta la eterna lucha entre la luz y la oscuridad, en un contexto de acción y estrategia entre fuerzas opuestas sacadas del mito y la leyenda. El lado de la luz y el lado oscuro son iguales en números y equivalentes en fuerza, pero no idénticos, su objetivo sin embargo es el mismo, conquistar los cinco puntos de poder o eliminar a la facción opuesta.

Las diferentes criaturas fantásticas bajo control de los jugadores forman 18 por bando de 8 clases diferentes. Ninguna de las suyas son iguales a las de su oponente.

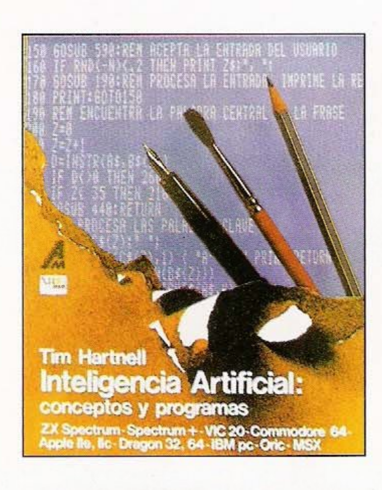
Archon se juega en dos pantallas. En una (la estratégica) los jugadores alternan turno moviendo sus piezas a posiciones favorables. Cuando se coloca una pieza en un cuadro ocupado por una pieza enemiga, el juego pasa temporalmente a la segunda presentación, la arena de combate. En un clásico juego de tablero, el defensor es barrido de su casilla sin mayor discusión. En Archon no se admiten tranquilidades de ese tipo; si alguien quiere algo tiene que luchar por ello. Cuando intente entrar en un cuadro ocupado por el enemigo lo verá: el cuadro en disputa se expande hasta llenar la pantalla... entonces en palabras de cierto monstruo de piel naranja del comic ¡ES LA HORA DE LOS MAMPORROS!

ARCHON



DRO SOFT. Fundadores, 3. Madrid 28028





INTELIGENCIA ARTIFICIAL: CONCEPTOS Y PROGRAMAS

Autor: Tim Hartnell
Editor: Anaya
Páginas: 267
Precio: N.D.

Un tema tan manido en los últimos tiempos como la inteligencia artificial y los sistemas expertos da pie a que muchos sean quienes especulen y cuenten sus «batallitas»; pero pocos ofrecen una visión tan real y directa como la hallada en esta obra. Su contenido es polivalente para distintos ordenadores, pues las sentencias PRINT están a la orden del día y no existen complicaciones con pantallas gráficas. Por otro lado, el autor ha utilizado un BASIC «neutro», reconocible por gran número de ordenadores.

Aparte de los programas, las anécdotas y libres interpretaciones hacen que la lectura sea realmente disfrutada.

Un hecho se viene a demostrar aquí, la inteligencia artificial no se encuentra tan distante de la mano de un programador medio. Es más, los programas y rutinas que aparecen pueden servir como punto de partida para crear programas más específicos para aplicaciones determinadas.

La inteligencia artificial aborda aplicaciones tan diversas como la utilización en el juego de las tres en raya, haciendo que el programa aprenda qué estrategias conducen a la derrota y la victoria, convirtiendo al ordenador en un competidor virtualmente invencible.

La comprensión del lenguaje natural y la traducción automática proporcionan material de base para investigar los mecanismos que permiten la comunicación entre los seres vivos.

El famoso programa que hizo alguna vez pensar en la posibilidad de que el psiquiatra podría ser sustituido por un ordenador, para tratar determinado tipo de desequilibrios mentales, tiene su réplica en estas páginas, si bien se trata de una versión simplificada.

Los sistemas expertos, capaces de aprender en base a razonamientos lógicos y los conocimientos previamente acumulados, ocupan otra buena parte de la extensión del libro.

Cabe destacar que uno de los mayores alicientes que se encuentran en la lectura de los diferentes capítulos, es la meticulosidad con que se explican las rutinas que componen los programas, de tal forma que siempre es posible saber qué estamos queriendo conseguir a medida que tecleamos.

En resumen, inteligencia artificial: conceptos y programas, es una obra didáctica y eminentemente práctica.

LENGUAJE MAQUINA DEL COMMODORE 64

Autor: Ian Sinclair
Editor: Gustavo Gili
Páginas: 159
Precio: N.D.

Algunos piensan que es primo de *sir* **Clive,** el del **Spectrum,** pero él asegura que no es así. Este conocido autor, con gran experiencia acumulada en la preparación de libros informáticos de fácil comprensión, vuelve a la carga para intentar quitarle gravedad al aprendizaje del código máquina.

Especialmente indicado para cuantos quieren dar sus primeros pasos en la comunicación directa con el microprocesador del **Commodore 64** sin complicarse la vida.

Explica las notaciones binaria y hexadecimal y su conversión mutua por medio de programas en BASIC.

Los números negativos y su manejo en los ordenadores quedan claramente expuestos, aunque con brevedad. Este tema que es tan importante a la hora de programar en código máquina, suele quedar relegado en la mayoría de libros que abordan esta clase de programación.

Una vez explicados los modos de direccionamiento y el papel que juegan los registros y el acumulador, **Sinclair** nos ofrece una serie de ejemplos



prácticos en forma de rutinas cortas, que pueden ser llevadas a la práctica utilizando cargadores desde BASIC, empleando POKE, READ, DATA y el correspondiente bucle FOR ... NEXT.

Si después de realizar los ejemplos, sigues convencido de que esto es lo tuyo, **Sinclair** continua describiendo un ensamblador, que facilita la tarea de programar en código máquina. La versión descrita se llama **Mikro,** pero no difiere mucho de otros ensambladores comerciales.

El libro concluye con dos ejemplos. El primero nos muestra una rutina sonora, que produce un tono al pulsar una tecla. El segundo es un renumerador de programas escritos en BASIC.

Este libro constituye una buena herramienta para la iniciación al aprendizaje del código máquina.

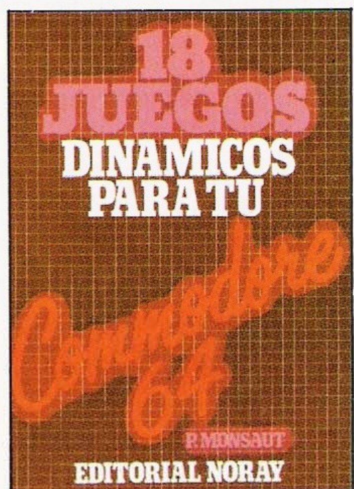
18 JUEGOS DINAMICOS PARA TU COMMODORE 64

Autor: P. Monsaut
Editor: Noray
Páginas: 95
Precio: 650

La mejor forma de aprender a programar parece ser que es programando. Pero resulta tedioso y complicado ser autodidacta. Es aquí donde toman protagonismo libros como éste. Con títulos tales como Slalom, Aterrizaje, La caza del pato, Cangrejos, Gran Premio, Tanque, etc., anima a los menos experimentados a iniciarse en base a programas relativamente breves, pero no por ello carentes de atractivo, como pudimos comprobar tecleándolos.

Los programas son una reproducción directa, al tamaño real, de la impresión matricial clásica, pero bastante cuidada.

Cabe resaltar que se ha huido de los caracteres gráficos de control diferido desde programa, para facilitar la tarea de copia, incluso los espacios en blanco aparecen indicados en número entre corchetes.



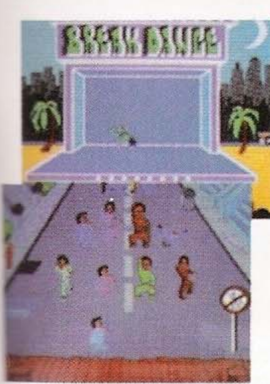
Para completar su fin, hubiera sido deseable que las distintas rutinas hubieran sido explicadas, aunque tal vez se pretenda que sea el propio lector quien los vaya descifrando a medida que avanza, pues bien es cierto que con un poco de interés y algún libro de referencia se terminan comprendiendo sin gran dificultad.

BREAKDANCE.™

BREAKIN' MADE EASY.

EPYX COMPUTER SOFTWARE

Strategy Games for the Action-Game Player



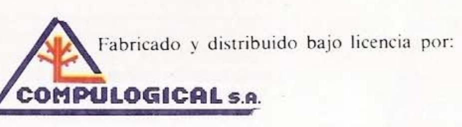
COMMODORE 64,
P. V. P.: 2.300 ptas.

La última locura de USA es el Break y no debe perdérselo. Ahora cualquiera puede bailar el Break. Sólo coja su joystick y controle a su bailarín de Break saltando (poping= hacer saltar como un tapón de champán), paseando como en la luna, estirándose y retorciéndose... todo en su pantalla de computador.

Breakdance, el juego, incluye un juego de acción en el que su bailarín trata de ganar a una pandilla de Breakers, un juego «como el simón», donde su bailarín tiene que duplicar los pasos del bailarín manipulado por el computador y la parte de baile libre donde usted desarrolla sus propias rutinas y el computador las realiza después para que las vea. Hay también un juego que le desafía a que realice la secuencia correcta de pasos para realizar un headspin (giro con cabeza) el suicidio u otros movimientos sin caerse.

¡Aprenda a bailar el Breakdance hoy! ¡Epyx lo hace fácil!

Uno o dos jugadores; control Joystick.





Santa Cruz de Marcenado, 31 - 28015 Madrid - Telef. 241 10 63

Distribuido en CATALUÑA y BALEARES por: DISCLUB, S. A.
Balmes, 58 08007 BARCELONA Teléfono (93) 302 39 08

SUMMER GAMES II.™

EIGHT NEW WAYS TO GO FOR THE GOLD.

EPYX™ COMPUTER SOFTWARE

Strategy Games for the Action-Game Player



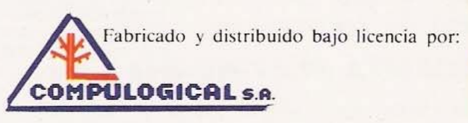
COMMODORE 64,
P. V. P.: 2.300 ptas.

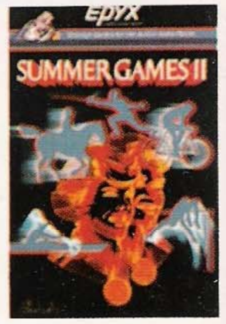
Seguro que Summer Games fue genial, pero ¿por qué pararse ahí? Deje que Summer Games II le lleve incluso más lejos con ocho nuevas competiciones, incluido: ciclismo, esgrima, kayaks, triple salto, remo, salto de altura, javalina e incluso equitación. Todos ellos pueden jugarse hasta con ocho jugadores y algunos como ciclismo, remo y esgrima le retan con una competición realista codo-a-codo.

Primero, decida cuál de los 18 países diferentes va a representar. Después, como en una verdadera competición olímpica, necesitará su propia estrategia y agilidad mental, no sólo velocidad y agilidad para superar cada prueba. Es tan realista, que hay incluso ceremonia de apertura y de cierre con presentación de medallas después de cada competición.

No es demasiado pronto para prepararse para 1988. Con la dieta oportuna, adecuado entrenamiento y horas de práctica, usted puede hacerlo. Mientras tanto, póngase su chandal, agarre su joystick y deje que Summer Games II le indique ocho nuevas formas de conseguir la medalla de oro.

Hasta ocho jugadores, control joystick.





Santa Cruz de Marcenado, 31 - 28015 Madrid - Telef. 241 10 63

Distribuido en CATALUÑA y BALEARES por: DISCLUB, S. A.
Balmes, 58 08007 BARCELONA Teléfono (93) 302 39 08